



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

**POROVNÁNÍ ČASOVĚ ZÁVISLÝCH METOD PRO
PŘEVOD BAREVNÉHO OBRAZU NA ŠEDOTÓNOVÝ**

COMPARISON OF TIME-DEPENDENT COLOR-TO-GRAY CONVERSIONS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. VLADIMÍR VLKOVIČ

VEDOUCÍ PRÁCE

SUPERVISOR

Doc. Ing. MARTIN ČADÍK, Ph.D.

BRNO 2018

Vysoké učení technické v Brně - Fakulta informačních technologií

Ústav počítačové grafiky a multimédií

Akademický rok 2017/2018

Zadání diplomové práce

Řešitel: **Vlkovič Vladimír, Bc.**

Obor: Počítačová grafika a multimédia

Téma: **Porovnání časově závislých metod pro převod barevného obrazu na šedotónový**

Comparison of Time-Dependent Color-to-Gray Conversions

Kategorie: Zpracování obrazu

Pokyny:

1. Seznamte se s problematikou časově závislých metod pro převod videa do stupní šedi.
2. Do existujícího systému TMS implementujte alespoň tři nové metody.
3. Navrhněte metodiku pro porovnání a hodnocení vlastností těchto metod uživatelským experimentem.
4. Pomocí navržené metodiky proveďte porovnání alespoň šesti vybraných metod. Pro ilustraci použijte alespoň jednu klasickou metodu (časově nezávislou).
5. Vyhodnoťte dosažené výsledky a diskutujte možnosti budoucího vývoje.

Literatura:

- Dle pokynů vedoucího.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese <http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Čadík Martin, doc. Ing., Ph.D.,** UPGM FIT VUT

Datum zadání: 1. listopadu 2017

Datum odevzdání: 23. května 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav počítačové grafiky a multimédií
L.S. 612 66 Brno, Božetěchova 2



doc. Dr. Ing. Jan Černocký
vedoucí ústavu

Abstrakt

Táto diplomová práca sa zaoberá porovnaním časovo závislých metód prevodu videa do šedotónu pomocou používateľského experimentu. Metodika testovania je založená na párovom teste pomocou metódy 2AFC. Obsahuje 2 varianty testov: test s referenčným videom a test bez referenčného videa. Pri analýze výsledkov je použitý koeficient zhody a koeficient konzistentnosti a korelačný koeficient. Používateľského testu sa zúčastnilo 60 ľudí ktorý vykonali 7200 párových porovnaní videí. Z testu bolo zistené, že najuniverzálnejšou metódou je časovo závislá metóda Hu14 a takisto bolo zistené, že niekoľko časovo nezávislých metód sa dostalo pred závislú metódu Kim09. Bol zaznamenaný čiastočný vplyv výberu vstupného videa na výkonnosť metód. Bola zistená korelácia medzi testom bez referencie a testom s referenciou. Hlavným prínosom tejto práce, že aj metódy určené pre obrázky dokážu byť za istých okolností konkurencieschopné v rámci metód určených pre video. Ďalším prínosom je zistenie, že pridanie referenčného videa ovplyvnilo rozhodovanie účastníkov len v malej miere.

Abstract

This masters thesis is focused around the comparison of time dependent video grayscale conversion methods based on a user experiment. The test methodology is based on the pairwise comparison method 2AFC. It is composed of two test variants: a test with a reference video and a test without a reference video. The coefficients of agreement, consistency and correlation are utilized in the result analysis. The testing was done on 60 subjects which have done 7200 pairwise comparisons. The test results show that the time dependent method Hu14 is the most universal. Time dependent method Kim09 was bested by some non-dependent methods. The results also indicate that there is some correlation between the two test variants and that the choice of the input video can have impact on the method performance. The main contribution of this thesis is that non-dependent methods can under certain circumstances rival the performance of dependent methods and that the addition of a reference video did not have a meaningful impact on the test subjects judgements.

Klíčové slová

prevod videí do šedotónu, 2AFC, temporálna koherencia, párové porovnávanie, konverzia do šedotónu, časová závislosť

Keywords

video grayscale conversion, 2AFC, temporal coherency, pairwise comparison, time dependency

Citácia

VLKOVIČ, Vladimír. *Porovnaní časově závislých metod pro převod barevného obrazu na šedotónový*. Brno, 2018. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Doc. Ing. Martin Čadík, Ph.D.

Porovnání časově závislých metod pro převod barevného obrazu na šedotónový

Prehlásenie

Prehlasujem, že som túto diplomovú prácu vypracoval samostatne pod vedením pána Doc. Ing. Martina Čadíka PhD.. Uviedol som všetky literárne pramene z ktorých som čerpal.

.....
Vladimír Vlkovič
22. mája 2018

Podakovanie

Ďakujem vedúcemu práce Doc. Ing. Martinovi Čadíkovi PhD., za vedenie mojej práce, cenné návrhy, rady a podnety .

Obsah

1	Úvod	3
2	Teória	4
2.1	Farebné modely	4
2.1.1	RGB	4
2.1.2	CIE XYZ 1931	4
2.1.3	CIE Luv 1976	5
2.1.4	CIE L*a*b* 1976	5
2.1.5	CIE L*C*h	5
2.2	Psychofyzikálne metódy	6
2.2.1	Nastavovacie metódy	6
2.2.2	Rozhodovacie metódy	6
2.3	Prehľad časovo závislých metód	7
2.3.1	Real-time Decolorization using Dominant Colors (2014)	7
2.3.2	Video decolorization Using visual proximity coherence optimization (2017)	9
2.3.3	Robust Color-to-gray via Nonlinear Global Mapping (2009)	10
2.4	Prehľad hodnotiacich štúdií pre prevod videa	12
2.4.1	Assessment of video tone-mapping (2013)	12
2.4.2	Evaluation of Tone Mapping Operators for HDR-Video (2013)	13
2.4.3	Perceptual Evaluation of Color-to-Grayscale Image Conversions (2008)	13
2.5	Temporálne artefakty	14
2.5.1	Globálne blikajúce artefakty	14
2.5.2	Lokálne blikajúce artefakty	14
2.5.3	Temporálny šum	15
2.6	Štatistika	15
2.6.1	Preferenčná matica	15
2.6.2	Kendallov koeficient zhody	16
2.6.3	Koeficient konzistencie	17
2.6.4	Významnosť	17
2.6.5	Pearsonov korelačný koeficient a koeficient determinácie	18
3	Návrh	19
3.1	Dataset	19
3.2	Metodika testovania	23
4	Implementácia	25
4.1	Tone mapping studio	25

4.2	TMOvideo a TMOv	26
4.3	Real-time Decolorization using Dominant Colors	27
4.4	Robust Color-to-gray via Nonlinear Global Mapping	28
4.5	Video decolorization Using visual proximity coherence optimization	28
4.6	Použité knižnice	29
4.6.1	OpenCV	29
4.6.2	matrix.h	29
4.6.3	Dlib	29
4.6.4	Pallas Solver	29
5	Testovanie	30
5.1	Základné údaje	30
5.2	Výsledky pre celý dataset	30
5.3	Výsledky pre jednotlivé videá	31
5.3.1	Výber zaujímavých výsledkov	33
5.4	Výsledky dotazníka	35
5.5	Zhrnutie	36
6	Záver	37
	Literatúra	38
A	Grafy	40
B	Odkazy na videá	45

Kapitola 1

Úvod

Video je možné definovať ako sled statických snímok idúcich za sebou. Tento sled spôsobuje dojem pohybu vďaka biologickým vlastnostiam ľudského oka ktoré má určitú zotrvačnosť.

V počiatkoch filmu, teda videa, boli filmy čiernobiele a nemé. Neskôr sa k čiernobielym filmom pridal aj zvuk. Prvý farebný film sa objavil zhruba pred 70 rokmi. Avšak farebný film sa nestal okamžite obľúbeným. Nebolo to len z dôvodu väčšej technickej náročnosti záznamu ale napočudovanie aj z dôvodu výrazu, pretože čiernobiely záznam mnohokrát lepšie vyjadroval požadovanú atmosféru. V niektorých krajinách sa natáčali filmy čiernobiely až do 70. rokov 20. storočia.

Čiernobiele video má svoje miesto aj v súčasnosti čoho dôkazom je možnosť zachytiť čiernobiele video na kamerách a mobilných zariadeniach. Dokonca existuje niekoľko príkladov z moderného filmového priemyslu kde bol čiernobiely záznam použitý ako napríklad Shindlerov zoznam alebo The Artist. Toto všetko je dôkazom toho, že prevod farebného videa na čiernobiele je stále relevantný a má budúcnosť.

Tému práce som si zvolil z dôvodu, že počas štúdia som v jednom projekte implementoval metódu na prevod obrázkov do šedotónu. Práca na tomto projekte bola zaujímavá a chcel som v nej nejakým spôsobom pokračovať. V dobe výberu zadania som mal skúsenosť len s touto jednou metódou no fascinovalo ma, že existuje celé odvetvie počítačovej grafiky ktoré produkuje rôzne spôsoby na riešenie tejto na prvý pohľad triviálnej problematiky.

Cieľom tejto práce porovnať a otestovať časovo závislé metódy pre prevod videa do šedotónovej reprezentácie. Súčasťou práce je implementácia týchto metód a vykonanie používateľského experimentu na navrhnutom datasete pomocou navrhnutej metodiky.

Text tejto práce je logicky štrukturovaný do piatich kapitol. Po úvode nasleduje kapitola zameraná na teóriu (2). V tejto kapitole sú uvedené teoretické základy ohľadom chýb v obraze, testovania a popis implementovaných metód. Nasledujúcou kapitolou je návrh (3). V kapitole sa rozoberá návrh datasetu určeného pre testovanie ako aj návrh samotnej metodiky testovania. Kapitola (4) po návrhu sa sústreďí na implementáciu metód popísaných v teórii ako aj na popis implementovanej knižnice pre načítanie videa do systému TMS. Po implementácii nasleduje kapitola testovanie (5). V tejto kapitole sa popisuje a vyhodnocuje používateľský experiment ktorý sa vykonal v rámci tejto práce. Poslednou kapitolou je záver (6) v ktorom sa zhrnú základné fakty o práci a jej výsledky.

Kapitola 2

Teória

Táto kapitola popisuje teoretický základ pre ďalší text práce. Ako prvý je uvedený prehľad farebných modelov. Nasledujúca podkapitola popisuje psychofyzikálne metódy. Popis časovo závislých metód je popísaný v ďalšej podkapitole. Nasleduje prehľad hodnotiacich štúdií pre prevod videa. Nasledujúca podkapitola popisuje typy temporálnych artefaktov. V závere kapitoly sú popísané štatistické výpočty potrebné k vyhodnoteniu testovania.

2.1 Farebné modely

Svetlo ktoré človek vníma predstavuje viditeľnú časť elektromagnetického spektra. Viditeľná časť elektromagnetického spektra sa nachádza v oblasti vlnových dĺžok 380 až 720 nm. V rámci tejto oblasti vnímame žiarenie ako farbu [7]. Farebný model predstavuje reprezentáciu farby v počítači.

2.1.1 RGB

RGB sa skladá z troch základných farieb: červená (red), zelená (green), modrá (blue). Všetky ostatné farby sú tvorené kombináciou týchto základných farieb. Farby sa dajú vyjadriť trojicou (farebným vektorom), ktorej zložky môžu mať hodnoty v intervale $< 0, 1 >$. Ak je každá zložka kódovaná jedným bytom potom sa môžu hodnoty uvádzať v celočíselnom rozsahu $< 0, 255 >$. Počet farebných odtieňov ktoré sa dajú reprezentovať je $256^3 = 16777216$ [7]. Model RGB patrí medzi najpoužívanějšíe modely, môže sa vyskytovať aj v tvare BGR.

2.1.2 CIE XYZ 1931

Tento priestor predpokladá, že každú farbu je možné definovať váženým súčtom primárnych farieb. Hodnoty jednotlivých zložiek sa pohybujú v rozmedzí $< 0, 1 >$ [7]. Spolu s týmto modelom bol zavedený aj chromatický diagram obr. 2.1. Z tohto diagramu je možné určiť farebný rozsah (color gamut) nejakého farebného priestoru vid. trojuholník na obrázku 2.1. Všetky vnútorné body tohto trojuholníka je možné vytvoriť kombináciou farieb v jeho vrcholoch [7].

2.1.3 CIE Luv 1976

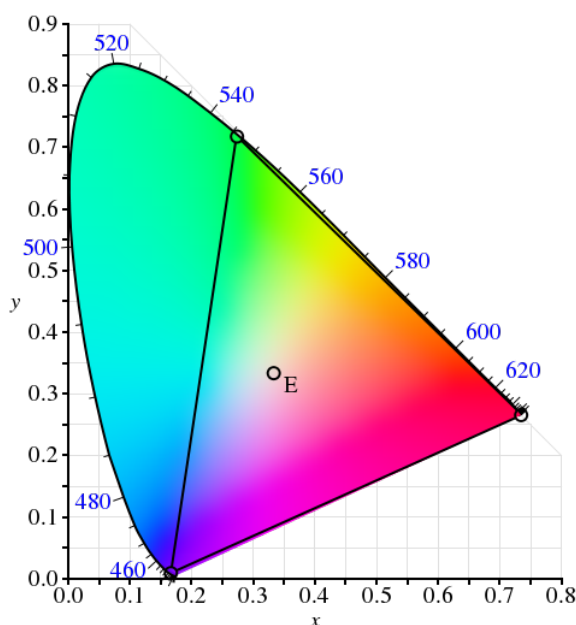
Model Luv je odvodený od modelu XYZ s tým rozdielom, že farby sú rozložené rovnomernejšie než v modeli XYZ. Rovnomernejšie rozloženie znamená, že priestorová vzdialenosť v tomto priestore bližšie odpovedá subjektívne vnímanému rozdielu medzi intenzitami dvoch farieb [7]. Z tohto dôvodu sa tento priestor považuje za perceptuálne uniformný. Zložka L predstavuje svetlosť a môže mať hodnotu v rozsahu $< 0, 100 >$, zložky u a v môžu nadobúdať hodnoty z intervalu $< -134, 220 >$ resp. $< -140, 122 >$ [6].

2.1.4 CIE $L^*a^*b^*$ 1976

Model býva často označovaný zjednodušene Lab alebo CIELAB. Tento model je ako model Luv odvodený od modelu XYZ. Popis a rozloženie farieb presnejšie odpovedá ľudskému vnímaniu farieb [7]. Zložka L^* je svetlosť v rozmedzí $< 0, 100 >$. Zložky a^* a b^* reprezentujú rozsahy farieb zelená až červená a modrá až žltá v rozsahy $< -128, 127 >$ [6]. Jednotlivé zložky modelu môžu byť nastavované individuálne, čo umožňuje zmeniť tón obrazu bez toho aby došlo k zmene jeho jas [7].

2.1.5 CIE L^*C^*h

Často býva označovaný len ako LCH. Model je podobný modelu Lab a takisto rozloženie farieb odpovedá ľudskému vnímaniu farieb. Zložka L^* reprezentuje jas (lightness), C^* je sýtosť (chroma) a h je odtieň (hue). Rozsah zložiek L^* a C^* je $< 0, 100 >$, zložka h môže nadobúdať hodnotu z intervalu $< 0, 360 >$ [11].



Obr. 2.1: CIE chromatický diagram

2.2 Psychofyzikálne metódy

Psychofyzika je veda zaoberajúca sa vzťahom medzi fyzickými podnetmi a pocitmi. Zaoberá sa testovaním ľudských zmyslov a reakciou na rôzne podnety.

Psychofyzikálne metódy sú nástrojom na meranie vnímania určitého podnetu a na vyhodnotenie výkonu používateľa. Používajú sa na odhalenie základného procesu vnímania aby bolo možné posúdiť výkonnosť používateľa [1].

Existujú dva typy metód: nastavovacie (adjustment) a rozhodovacie (judgement). Tieto dva typy sa považujú za protiklady. Na základe typu podnetu môžu byť metódy rozdelené na podúlohy [1]:

- detekcia (detection) - používateľ rozhoduje či podnet bol prítomný alebo nie
- diskriminácia (discrimination) - používateľ vyberá podnet v porovnaní z iným podnetom
- klasifikácia (classification) - používateľ vyberá podnet z viacerých kategórií

2.2.1 Nastavovacie metódy

Pri tomto type metód je používateľovi poskytnutý podnet ktorý musí nastaviť tak aby splnil kritérium zadané experimentátorom. Metóda je subjektívna pretože je závislá na správnom pochopení zadaného kritéria pretože, pri zlom vysvetlení alebo nepochopení používateľa nastáva skreslenie výsledkov. Používatelia hodnotia túto metódu ako zábavnejšiu oproti metóde rozhodovacieho typu [1].

Medzi najznámešiu metódu tohto typu patrí *prah* (threshold). Táto metóda je aplikovateľná viacerými spôsobmi pre ilustráciu uvedieme metódu limitov (method of limits) ktorá má vzostupnú a zostupnú variantu. Pri vzostupnej variante je hodnota podnetu nastavená veľmi nízko. Túto hodnotu následne zvyšujeme a čakáme kedy používateľ zareaguje na podnet. Zostupná varianta funguje opačne [18].

Pri metóde *matching* sú používateľovi zobrazené dva podnety. Následne je používateľovi povedané aby nastavil jeden podnet tak aby sa čo najviac podobal druhému podnetu [1].

Nulling sa považuje za najsilnejšiu metódu spomedzi nastavovacích metód. Používateľovi je zobrazený skreslený podnet, následne je používateľ požiadaný aby pomocou manipulácie s nastaveniami skreslenie odstránil. Skreslenie podnetu býva jednoznačné takže je pre užívateľa pomerne jednoduché určiť kedy skreslenie odstránil [1].

2.2.2 Rozhodovacie metódy

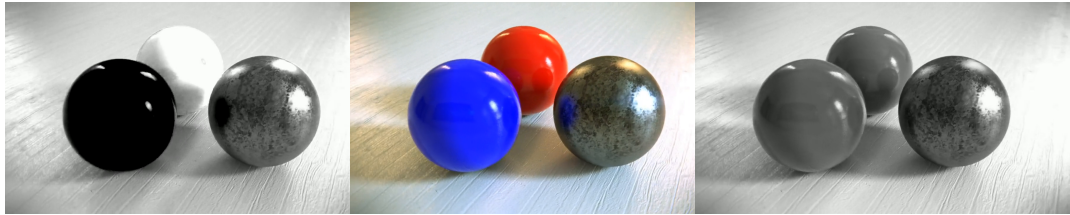
Pri rozhodovacích metódach je používateľovi prezentovaný podnet ktorý je požiadaný klasifikovať. Prezentované podnety sú v réžii experimentátora a používateľ len urobí rozhodnutie na základe podnetu. Používatelia považujú rozhodovacie metódy za únavné pretože tieto experimenty trvajú dlhší čas [1].

Najjednoduchšou a najznámejšou metódou je *yes-no* (áno-nie). Metóda sa používa zväčša pri detekcii ale môže byť použitá aj pri diskriminácii za istých podmienok. Otázka od experimentátora môže byť položená nasledovne: Videli ste podnet? Na túto otázku môže používateľ odpovedať len dvomi spôsobmi a to áno alebo nie [1].

Ďalšou metódou je *rating scale* alebo v preklade stupnica. Pri tejto metóde je používateľovi na rozhodovanie poskytnutá stupnica začínajúca možnosťou videl som podnet a končiacou možnosťou nevidel som podnet. Medzi týmito možnosťami môže byť variabilné

množstvo ďalších možností (typicky dohromady 5) alebo je prípustná aj kontinuálna stupnica s nie presne vyhranenými možnosťami [1].

Celý názov metódy je *Two-alternative forced choice* alebo v preklade nútený výber z dvoch alternatív. V tejto metóde sú používateľovi prezentované práve dva podnety a musí vybrať jeden z nich [1]. Podnety môžu byť prezentované postupne alebo vedľa seba. Prípadne môže byť pridaný podnet kvôli referencii. Príklad kedy musí používateľ vybrať šedotónový obraz ktorý je bližší referenčnému je na obr. 2.2 Metóda sa radí do skupiny rozhodovacích metód ktoré majú štruktúru typu zobrazenie podnetu a následne rozhodnutie. Oproti iným psychofyzikálnym metódam poskytuje viac objektivity [18].



(a) Obraz 1

(b) Referenčný obraz

(c) Obraz 2

Obr. 2.2: Ukážka párového testu s referenčným obrazom.

2.3 Prehľad časovo závislých metód

V podkapitole sú rozobraté 3 časovo závislé metódy pre prevod videa do šedotónovej reprezentácie. Za časovo závislú metódu sa považuje taká ktorá pri prevode do šedotónu využíva aj predchádzajúce snímky videa nie len aktuálne prevádzaný snímok. Metódy ktoré prevádzajú len aktuálny snímok bez informácií o predchádzajúcich snímkoch sa nazývajú časovo nezávislé.

2.3.1 Real-time Decolorization using Dominant Colors (2014)

Metóda bude ďalej v práci označovaná podľa autora ako Hu14. Základ metódy spočíva v získaní deskriptora dominantných farieb(ďalej DCD) z obrazu. Zo vstupného obrazu sa získa obraz hrán(obr. 2.3). Do tohto obrazu hrán sa pridajú farby a tým vznikne tzv. I-edge(obr. 2.3). Farby v I-edge sú potom prevedené do farebného priestoru CIE LUV (2.1.3). Z obrazu I-edge je následne získaný histogram farieb. Z tohoto histogramu sa vytvorí DCD zhľukovaním perceptuálne podobných farieb. Rozdielnosť farieb je definovaná ako $d = |v_1 - v_2|$, kde v reprezentuje farbu. Farby sú klasifikované ako podobné ak d je menšie ako prah δ . Ak je táto podmienka splnená je vytvorená nová farba v_{new} ktorá nahradí originálne farby a je vygenerovaná nasledovne [5]:

$$v_{new} = \frac{v_1 p_1 + v_2 p_2}{p_1 + p_2} \quad (2.1)$$

kde p je percentuálny výskyt farby.

Následne je DCD konvertovaný do šedotónových farieb pomocou setu 66 váh (obr. 2.3). Tieto váhy sú uniformne rozmiestnené. Odfarbením získame maticu G ktorá je definovaná

tak, že obsahuje všetky páry p_i, g_i , kde g_i je prevedená šedotónová farba a p_i je jej percentuálny podiel.[5]

Ďalej je zadefinovaná metrika na meranie kvality prevodu:

$$\xi(G) = \sum_{i,j=1}^N H_{i,j}[\lambda K_{i,j} + (1 - \lambda)D_{i,j}] \quad (2.2)$$

kde

$$K_{i,j} = \begin{cases} 1 & \text{if } |g_i - g_j| > \tau \\ 0 & \text{otherwise} \end{cases} \quad (2.3)$$

$$D_{i,j} = |g_i - g_j| \quad (2.4)$$

$$H_{i,j} = |v_i - v_j|(p_i + p_j) \quad (2.5)$$

Váhy ktoré dosiahnu najnižšie skóre sú považované za najlepšieho kandidáta ako decolorizovať obraz.[5]

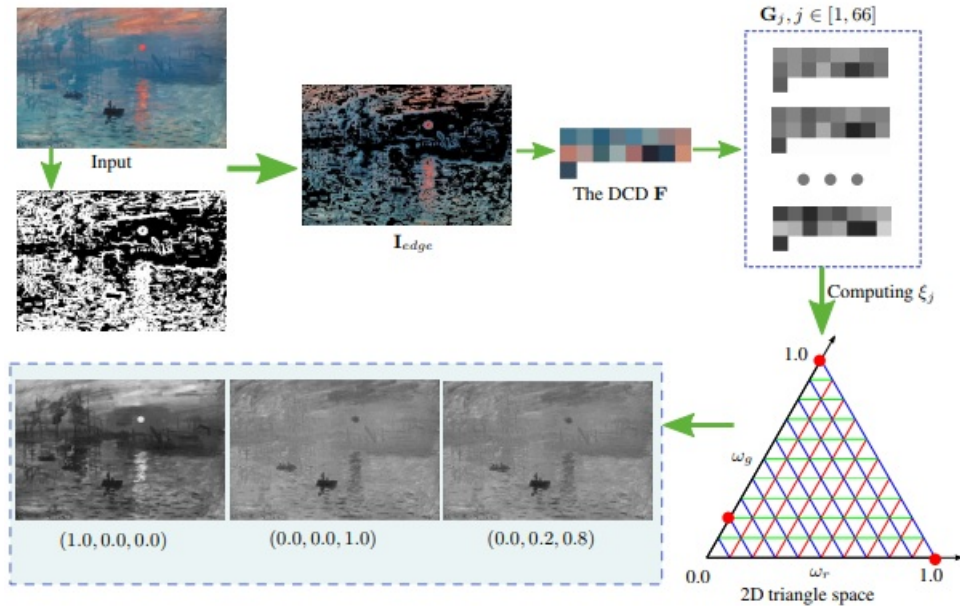
Pri dekolorizácii videa sa počíta s aktuálnym snímkom a predošlým snímkom. Použije sa vypočítaný histogram(H_{i-1}) resp. DCD snímku $i - 1$ potom sa vypočíta histogram(H_i) pre aktuálny snímok i . Vypočíta sa rozdiel d_i medzi histogramami H_i a H_{i-1} , pričom $d_i \in [0, 1]$. Následne je vypočítaný faktor zmeny c_i pomocou funkcie ktorá je definovaná:

$$filter(x, \theta, \gamma) = \begin{cases} \theta(x/\theta)^\gamma & \text{if } x < \theta \\ 1 + (\theta - 1)(\frac{1-x}{1-\theta})^\gamma & \text{if } x \geq \theta \end{cases} \quad (2.6)$$

kde x je rozdiel histogramov, $\gamma = 1.6$ a $\theta = 0.02$ sú kontrolné parametre.[5]

Potom sa vyhladá váha W_{close} ktorá je najbližšia váhe W_{i-1} predošlého snímku. Následne sa váha W_{close} modifikuje faktorom zmeny c_i a to nasledovne [5]:

$$W_i = W_{i-1} + c_i \times (W_{close} - W_{i-1}) \quad (2.7)$$



Obr. 2.3: Pipeline dekolorizačného algoritmu[5].

2.3.2 Video decolorization Using visual proximity coherence optimization (2017)

Metóda bude ďalej v práci označovaná podľa autora ako Tao17. Na začiatku sa jednotlivé snímky prevedú do Lab 2.1.4 farbeného priestoru následne sa vypočíta odlišnosť snímok (proximity computation) v chromatickej a jasovej zložke. Vypočítané odlišnosti slúžia s ako vstupné parametre pre klasifikátor ktorý určí ktorý z troch dekolorizačných algoritmov sa má použiť. Tieto algoritmy sú High-proximity decolorization strategy (ďalej HPD), Median-Proximity Decolorization (ďalej MPD) a Low-Proximity Decolorization (ďalej LPD).

Pri malej rozdielnosti snímok sa použije HPD. Najskôr sa vypočíta differential refinement frame D_i ktorý sa následne pripočíta k prechádzajúcemu šedotónovému snímku G_{i-1} aby sme dostali aktuálny šedotónový snímok G_i . Výpočet D_i vyzerá nasledujúco:

$$D_i(x) = \phi \sum_{k \in a, b} \left(I_i^k(x + v) - I_{i-1}^k(x) \right) + (1 - \phi) \left(I_i^L(x + v) - I_{i-1}^L(x) \right) \quad (2.8)$$

kde ϕ je váha pre L zložku modelu Lab, x je pohybový vektor pixelu a I je farebný snímok.

Pri väčšej rozdielnosti snímok je použitý algoritmus MPD. V tomto algoritme sa počíta coherence refinement frame C_i . Pri definícii C_i sa používa metrika:

$$M_i = \|I_i(x) - I_{i-1}(x - v)\|_2, \quad (2.9)$$

kde $\| \cdot \|$ je L-2 norma vektora, v je pohybový vektor pixelu.

Definícia C_i je inšpirovaná Markov random field modelom kde pravdepodobnosť C_i je difonovaná nasledovne:

$$P(C_i) \propto \prod_{x \in I_i} \exp\left\{-(C_i(x) - G_{i-1}(x))/\sigma_p^2\right\} \times \exp\left\{-\frac{1}{M_i(x)}(C_i(x) - G_{i-1}(x - u))/\sigma_t^2\right\} \quad (2.10)$$

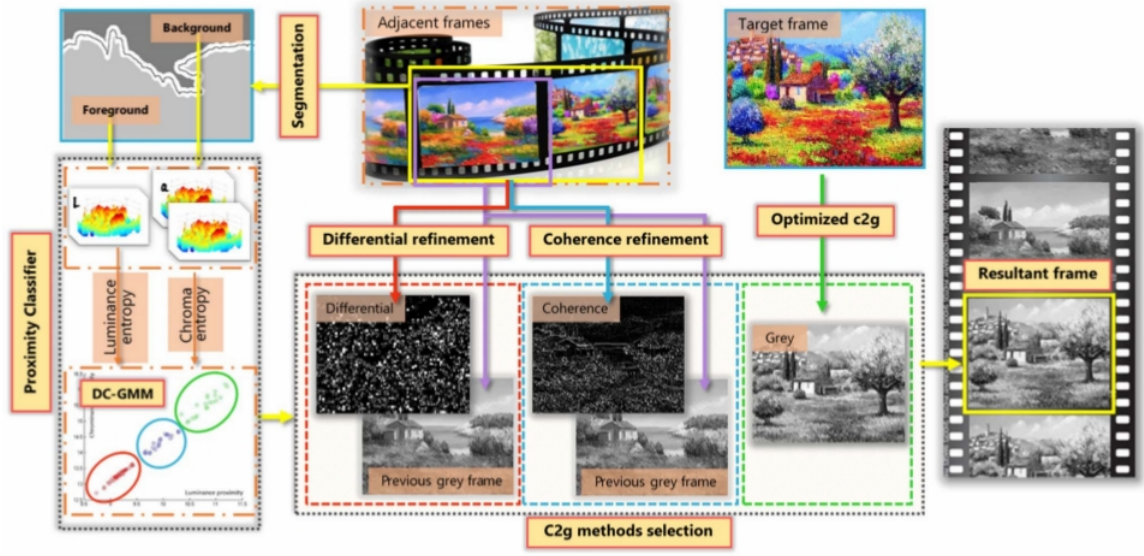
Maximalizáciou tejto pravdepodobnosti dosiahneme výsledný coherence refinement frame pomocou ktorého dostaneme aktuálny šedotónový snímok G_i ako váženú sumu G_{i-1} a C_i . Pri veľkom rozdieli snímok sa použije Low-Proximity Decolorization Strategy a to tak, že sa minimalizuje energia daná rovnicami:

$$E_s = \sum_{(x, y) \in I} ((G_i(x) - G_i(y)) - \eta_i(x, y))^2 \quad (2.11)$$

$$E_t = \sum_{x \in I} ((G_i(x) - G_{i-1}(x)) - (I_i(x) - I_{i-1}(x)))^2 \quad (2.12)$$

$$E = (1 - \beta)E_s + \beta E_t \quad (2.13)$$

kde $\eta_i(x, y)$ je rozdiel medzi dvoma pixelmi, E_s je energia snímku I_i a E_t je energia medzi snímkami I_i a I_{i-1} . Energia E je spojením oboch energií s váhou β .



Obr. 2.4: Schéma algoritmu[17].

2.3.3 Robust Color-to-gray via Nonlinear Global Mapping (2009)

Metóda bude ďalej v práci označovaná podľa autora ako Kim09. Metóda používa globálne mapovanie na prevod obrazu do šedotónovej reprezentácie. Tento prístup zamedzuje nehomogénnej konverzii oblastí v ktorých je farba konštantná, čo by mohlo viesť k narušeniu vzľadu výsledného obrazu. Globálne mapovanie je špecifikované nasledovne[9]:

$$g(x, y) = L + f(\theta)C \quad (2.14)$$

kde L je hodnota jasnosti(lightness), C je sýtosť(chroma) a θ je uhol odieňu(hue) v farebnom priestore CIE LCH (2.1.5) na pixely (x, y) . [9] Funkcia f je definovaná nasledovne:

$$f(\theta) = \sum_{k=1}^n (A_k \cos(k\theta) + B_k \sin(k\theta)) + A_0 \quad (2.15)$$

kde A_k , B_k a A_0 sú neznáme parametre, a úlohou metódy je ich nájdenie. Tieto hodnoty sú ďalej spomenuté v rovnici.

Pre zachovanie rysov v šedotónovom obraze sa minimalizuje rozdiel gradientov pôvodného obrazu a výsledného šedotónového obrazu. Energia je definovaná nasledovne[9]:

$$E_s = \sum_{x, y \in \Omega} \|\nabla g(x, y) - G(x, y)\|^2 \quad (2.16)$$

kde (x, y) je pixel v obraze Ω , $g(x, y)$ je pixel v šedotónovom obraze, $\nabla g(x, y)$ je gradient šedotónového obrazu a $G(x, y)$ je rozdiel farieb pôvodného obrazu[9]. Gradient šedotónového obrazu je definovaný:

$$\nabla g(x, y) = (g(x + 1, y) - g(x - 1, y), g(x, y + 1) - g(x, y - 1)) \quad (2.17)$$

Rozdiel farieb c vstupného pixelu je:

$$G(x, y) = \begin{pmatrix} G^x(x, y) \\ G^y(x, y) \end{pmatrix}^T = \begin{pmatrix} c(x + 1, y) \ominus c(x - 1, y) \\ c(x, y + 1) \ominus c(x, y - 1) \end{pmatrix}^T \quad (2.18)$$

Počítanie rozdielu farieb je definované v nasledujúcom vzťahu:

$$c_i \ominus c_j = \text{sign}(c_i, c_j) \sqrt{\Delta L_{ij}^2 + \left(\alpha \frac{\sqrt{\Delta a_{ij}^{*2} + \Delta b_{ij}^{*2}}}{\Re} \right)^2} \quad (2.19)$$

kde L , a^* a b^* sú reprezentáciou farby c vo farebnom priestore CIE $L^*a^*b^*$ (2.1.4), \Re je normalizačná konštanta $\Re = 2.54\sqrt{2}$ [9]. Pre kompaktnú reprezentáciu sa použije vektorový tvar rovnice 2.15:

$$f(\theta) = t^T x, \quad (2.20)$$

kde $t = (t_i)$ a $x = (x_i)$ sú vektory o rozmeroch $(2n + 1 \times 1)$. Tieto vektory sú definované:

$$t_i = \begin{cases} \cos(i\theta) & \text{pre } 1 \leq i \leq n \quad (i = 1, 2, 3, 4) \\ \sin((i - n)\theta) & \text{pre } n + 1 \leq i \leq 2n \quad (i = 5, 6, 7, 8) \\ 1 & \text{pre } i = 2n + 1 (i = 9) \end{cases} \quad (2.21)$$

$$x_i = \begin{cases} A_i & \text{pre } 1 \leq i \leq n \quad (i = 1, 2, 3, 4) \\ B_{i-1} & \text{pre } n + 1 \leq i \leq 2n \quad (i = 5, 6, 7, 8) \\ A_0 & \text{pre } i = 2n + 1 (i = 9) \end{cases} \quad (2.22)$$

Množiny ktoré sú v zátvorke sú hodnoty i pre $n = 4$. Hodnota n bola autormi označená ako ideálna voľba. Vektor \hat{x}_{image} jednoznačne definuje funkciu f spomenutú v rovnici 2.14. Túto maticu je možné priamo použiť do rovnice 2.20. Vektor \hat{x}_{image} sa získa vyriešením rovnice:

$$\hat{x}_{image} = (M_s + \lambda I)^{-1} b_s, \quad (2.23)$$

kde horný index -1 označuje pseudoinverznú maticu a M_s a b_s sú definované nasledovne:

$$M_s = \sum_{\Omega} (uu^T + vv^T), \quad (2.24)$$

$$b_s = \sum_{\Omega} (pu + qv), \quad (2.25)$$

kde u a v sú vektory a p a q sú skaláry. Vektory u a v sú definované:

$$u = (C \cdot t)_x, \quad (2.26)$$

$$v = (C \cdot t)_y, \quad (2.27)$$

kde dolné indexy označujú parciálnu deriváciu v danom smere. Vektor t je definovaný v rovnici 2.21. Písmeno C označuje hodnotu z farebného modelu LCH (2.1.5). Hodnoty p a q sa vypočítajú týmto spôsobom:

$$p = G^x - L_x, \quad (2.28)$$

$$q = G^y - L_y, \quad (2.29)$$

kde G^x a G^y značia gradient v smere danom horným indexom a L_x a L_y je rozdiel jasovej zložky na osi určenej dolným indexom.

Pre dodržanie temporálnej koherencie vo videách je energia definovaná:

$$E_t = \{(g_c(x, y) - g_p(x, y)) - (c(x, y) \ominus c_p(x, y))\}^2, \quad (2.30)$$

kde dolný index c značí aktuálny snímok, dolný index p značí predchádzajúci snímok [9].

Podobne ako pri obrázku vektor \hat{x}_{video} jednoznačne definuje funkciu f a je ho možné priamo použiť do rovnice 2.20. Vektor \hat{x}_{video} sa získa vyriešením rovnice:

$$\hat{x}_{video} = (M_v + \lambda I)^{-1} b_v, \quad (2.31)$$

kde M_v a b_v sú definované ako:

$$M_v = (1 - \beta)M_s + \beta M_t, \quad (2.32)$$

$$b_v = (1 - \beta)b_s + \beta b_t, \quad (2.33)$$

kde β je používateľom zadaný parameter ktorý vyvažuje časovú(temporal) energiu E_t a priestorovú(spatial) energiu E_s . Ďalej M_t a b_t sú definované:

$$M_t = \sum_{\Omega} (C_c^2 t_c t_c^T), \quad (2.34)$$

$$b_t = \sum_{\Omega} C_c (g_p + (c_c \ominus c_p) - L_c) t_c, \quad (2.35)$$

kde t je definované v rovnici 2.21, C a L sú farebné zložky modelu LCH, c je farba a dolné indexy c a p označujú aktuálny a prechádzajúci snímok.

2.4 Prehľad hodnotiacich štúdií pre prevod videa

Hodnotiaca štúdia ohľadom prevodu videa do šedotónu pravdepodobne nebola realizovaná tak v tejto podkapitole sú popísané doteraz vykonané štúdie ohľadom tone mappingu HDR videa a prevodu obrázkov do šedotónovej reprezentácie. Tone mapping je proces mapovania HDR (vysoký dynamický rozsah) obrázkov do priestoru ktorý je možný zobrazit na súčasných monitoroch.

2.4.1 Assessment of video tone-mapping (2013)

Jedná sa o jednu z prvých štúdií ktorá používa HDR video sekvencie pri testovaní tone-mapping operátorov (ďalej TMO). Tento experiment funguje bez referenčného videa. TMO boli nastavované rôzne parametre a každé nové nastavenie sa považovalo za nový operátor. Výkonnosť TMO sa skúma pomocou ohodnocovacieho experimentu v ktorom sa používajú dve kritéria: celková kvalita a čo najvernejšie zobrazenie scény z reálneho sveta [15].

Pri hodnotení kvality bola úloha zadaná nasledovne: Najvyššou známku ohodnotte tie video klipy ktoré preferujete a vybrali by ste si ich do vášho video albumu. Pri vernosti zobrazenia bola úloha formulovaná: Ohodnotte video na základe toho ako veľmi je podobné tomu ako si predstavujete zobrazenie scény v skutočnosti. Na určovanie ohodnotenia bol nad každým video umiestnený slider s 11 dielikmi stupnice 2.2.2 (od veľmi slabé až po excelentné) [15]. Autori štúdie sa rozhodli pre ohodnotie pomocou stupnice preto, že párové porovnanie považovali pri počte metód ktoré mali za zdĺhavé.

Experimentu sa zúčastnilo 10 ľudí. Boli vyzvaní aby použili svoje okuliare ak majú a bola im položená otázka či majú normálne farebné videnie. Väčšina účastníkov experimentu vykonala experiment dva krát s jednodenným odstupom kvôli oddychu. Účastníci boli rozdelení do dvoch skupín a každej skupine bola pridelená jedna úloha [15].

2.4.2 Evaluation of Tone Mapping Operators for HDR-Video (2013)

V štúdiu bolo použitých 11 TMO špeciálne zameraných na videá. Boli vykonané dva experimenty a to: vyhodnotenie kvality a pomocou párového porovnania.

Experiment vyhodnotenia kvality bol kociťovaný ako ohodnotenie 6 HDR videí na ktoré boli aplikované všetky TMO. Piatim expertným účastníkom testu boli pustené videá v náhodnom poradí a museli zhodnotiť nasledujúce atribúty: celkový jas, celkový kontrast, celková sýtosť farieb, temporálna farebná konzistentnosť, blikanie videa. Tieto atribúty boli zvolené z dôvodu, že predstavujú najčastejšie problémy ktoré sa môžu vyskytnúť vo video sekvencii. Tento experiment poskytol zaujímavé informácie, avšak takéto priame ohodnotenie nie je dostatočne citlivé pre malé rozdiely medzi jednotlivými TMO [3].

V párovom porovnávaní bolo použitých 7 TMO. V každom pokuse boli účastníkovi experimentu zobrazené dve videá rovnakej HDR scény s tým, že na videá bol použitý iný operátor. Potom bol účastník požiadaný aby vybral to video ktoré podľa jeho predstavy zobrazuje scénu bližšie k skutočnosti. Vykonanie párového testu v plnom rozsahu bolo príliš časovo náročné, namiesto toho autori použili metódu Quicksort na zredukovanie porovnaní. Experimentu sa zúčastnilo 18 ľudí. Na vyhodnotenie testovania bola použitá metóda JND. Ak 75% používateľov vyberie jednu variantu namiesto druhej potom kvalitatívny rozdiel je 1 JND. Dáta sú štandardizované pomocou tzv. zákona porovnávacích rozhodnutí (law of comparative judgement) [3].

2.4.3 Perceptual Evaluation of Color-to-Grayscale Image Conversions (2008)

Táto štúdia sa zameriava na prevod farebných obrazov do šedotónovej reprezentácie. Bolo použitých 7 metód na prevod do šedotónu. V rámci štúdie bol vytvorený 24 obrázkový dataset. V datasete sú zastúpené obrazy rastlín, portréty, počítačovo generované obrazy, testovacie obrazy, ovocie a zelenina [20].

Na testovanie bola použitá metóda párových porovnaní, konkrétne metóda 2AFC vid **2.2.2**. Obrázky boli porovnávané každý s každým, čo je pri počte metód $n = 7$, $(n \times (n - 1))/2 = (7 \times 6)/2 = 21$ porovnaní. Celkové množstvo porovnaní dostaneme ako počet porovnaní \times počet videí v datasete, čo je 504 porovnaní. Tento počet je príliš vysoký pre jedného účastníka experimentu. Testovanie bolo z tohto dôvodu rozdelené na 3 sety po 8 obrázkov. Takto bolo docielené zníženie počtu porovnaní na 168 pre jedného účastníka a pilotnou štúdiou bolo potvrdené, že sa jedná o akceptovateľnú hodnotu kedy účastníci nepocitujú únavu a stratu koncentrácie. Boli vykonané dva experimenty a to: experiment s referenciou (presnosť) a experiment bez referencie (preferencia) [20].

Pri experimente s referenciou boli šedotónové obrázky prezentované vedľa originálneho (referenčného) farebného obrazu. Pri teste boli účastníci požiadaný vybrať obrázok ktorý bol vzhľadovo bližší k referenčnému obrázku. Konkrétne úloha pre účastníka testu bola formulovaná: Vašou úlohou je vybrať šedotónový obrázok ktorý viac odpovedá farbám na referenčnom obrázku [20].

V experimente bez referencie boli účastníkom zobrazené dva šedotónové obrázky vedľa seba bez akejkoľvek referencie. Úloha pre účastníkov bola definovaná nasledovne: Vašou úlohou je z prezentovaného páru obrázkov vybrať ten ktorý preferujete [20].

Dáta boli konvertované do štandardného skóre (z-score) pomocou zákona porovnávacích rozhodnutí (law of comparative judgement). Testu sa zúčastnilo 121 ľudí, z ktorých 60 urobilo experiment s referenciou a 59 bez referencie. Dvaja ľudia boli vyradení z dôvodu poruchy vnímania farieb [20]. Dataset z tejto štúdie sa stal veľmi populárnym na testovanie nových metód prevodu farebného obrazu do šedotónu. Používa sa na zrovnávanie výsledkov viacerých metód a stal sa v podstate benchmarkom.

2.5 Temporálne artefakty

V tejto kapitole sú popísané chyby obrazu ktoré môžu vzniknúť použitím metódy na prevod jednotlivých snímkov (frame-by-frame) video sekvencie.

2.5.1 Globálne blikajúce artefakty

Globálne blikajúce artefakty (Global flickering artifacts) sú charakterizované náhlými zmenami jasů v po sebe idúcich snímkoch video sekvencie (obr. 2.5c). Tieto artefakty sa objavujú z toho dôvodu, že metódy pri prevode adaptujú svoje mapovanie na základe štatistík o obraze ktoré bývajú nestabilné v časovej rovine [2].



(a) Snímka č. 233

(b) Snímka č. 234

(c) Snímka č. 235

Obr. 2.5: Ukážka globálneho blikajúceho artefaktu, zmena jasů je v celom obraze avšak najviac viditeľné je to na stene v pozadí.

2.5.2 Lokálne blikajúce artefakty

Lokálne blikajúce artefakty (Local flickering artifacts) sú definované podobne ako globálne avšak s tým rozdielom, že vznikajú na obmedzenej ploche (obr. 2.6). Prevažne vznikajú vtedy ak metóda mapuje pixel na základe jeho okolia. Malé zmeny v tomto okolí v po sebe idúcich snímkoch môžu viesť na odlišné mapovanie pixelu [2].



(a) Snímka č. 55

(b) Snímka č.56

Obr. 2.6: Ukážka lokálne blikajúceho artefaktu.

2.5.3 Temporálny šum

Temporálny šum (Temporal noise) je pomerne bežným artefaktom ktorý sa vyskytuje v digitálnych video sekvenciách (obr. 2.7b). Najviditeľnejší šum vzniká pri nedostatočných svetelných podmienkach. Na obrázkoch má šum menší dopad na subjektívne vnímanie kvality obrázku. Pri video sekvencii je oveľa viac rušivý, kvôli tomu, že je premenlivý v čase [2].



(a) Snímka č. 10

(b) Snímka č.11

Obr. 2.7: Ukážka temporálneho šumu.

2.6 Štatistika

Pre vyhodnotenie dát nazbieraných v testovaní je ich potrebné nejakým spôsobom analyzovať. V tejto kapitole sú popísané štatistické spôsoby analýzy dát nazbieraných párovým testovaním.

2.6.1 Preferenčná matica

Pri párovom porovnávaní (2.2.2, 2AFC) sa na záznam rozhodnutí jednotlivých účastníkov pre jednotlivé videá/obrázky používa preferenčná matica. Táto matica má rozmery $t \times t$, kde t je počet operátorov použitých na video alebo obrázok [13].

Interpretáciu matice si ukážeme nasledujúcim príkladom: V tab. 2.1 v bunke s riadkom Lee10 a stĺpcom CIE Y je zapísané číslo 1, čo znamená, že účastník testu považoval video odfarbené operátorom Lee10 za príťažlivejšie v porovnaní s operátorom CIE Y. Na hlavnej diagonále nie sú žiadne dáta pretože neporovnávame rovnaké operátory medzi sebou. Pre výpočet preferenčnej matice pre video alebo obrázok sa spočítajú preferenčné matice všetkých účastníkov ktorý hodnotili konkrétne video (tab. 2.2).

–	CIE Y	Decolorize07	Hu14	Kim09	Lee10	Zheng15	Skóre
CIE Y	–	0	1	1	0	0	2
Decolorize07	1	–	1	1	0	1	4
Hu14	0	0	–	0	1	0	1
Kim09	0	0	1	–	0	1	2
Lee10	1	1	0	1	–	0	3
Zheng15	1	0	1	0	1	–	3

Tabuľka 2.1: Príklad preferenčnej matice jedného účastníka testu.

–	CIE Y	Decolorize07	Hu14	Kim09	Lee10	Zheng15	Skóre
CIE Y	–	8	7	9	9	4	37
Decolorize07	2	–	3	4	6	3	18
Hu14	3	7	–	7	9	5	31
Kim09	1	6	3	–	7	3	20
Lee10	1	4	1	3	–	1	10
Zheng15	6	7	5	7	9	–	34

Tabuľka 2.2: Príklad preferenčnej matice pre jedno video.

2.6.2 Kendallov koeficient zhody

V anglickom znení Kendall coefficient of agreement je koeficient ktorý určuje ako veľmi sa jednotliví účastníci zhodli medzi sebou v rámci testu. Pre jeho zadefinovanie potrebujeme poznať hodnotu Σ , ktorá je definovaná nasledovne [13]:

$$\Sigma = \sum_{i \neq j} \binom{p_{ij}}{2}, \quad (2.36)$$

kde p_{ij} je počet prípadov kedy bol operátor op_i preferovanejší pred operátorom op_j . Suma sa počíta pre $t(t-1)$, kde t je počet operátorov ako už bolo spomínané. Σ je suma počtu zhôd medzi jednotlivými párami [13].

Koeficient zhody v rámci účastníkov testu je definovaný:

$$u = \frac{2\Sigma}{\binom{t}{2}\binom{s}{2}} - 1, \quad (2.37)$$

kde s je počet účastníkov. Ak by všetkých s účastníkov urobilo počas testu identické rozhodnutia potom by bola hodnota u rovná 1. Čím bude menšia zhoda medzi účastníkmi, tým bude hodnota u menšia. Najmenšia hodnota ktorú môže u nadobudnúť je $-1/(s-1)$. Pre každé video alebo obrázok je možné vypočítať koeficient zhody čo nám umožní lepšiu predstavu o podobnosti rozhodnutí medzi jednotlivými účastníkmi testu [13].

2.6.3 Koeficient konzistencie

V párovom porovnávaní je jedným z dôležitých faktorov konzistencia alebo inak povedané tranzitívnosť. Konzistenciu najlepšie vysvetlíme na nasledujúcom príklade. Ak vyhodnocujeme tri objekty (operátory) označme ich A , B a C . Ak účastník testu vyjadrí svoje rozhodnutie ako $A \rightarrow B$ (šípka značí, že A je pre účastníka prijateľnejšie ako B), $B \rightarrow C$ a $A \rightarrow C$ potom považujeme jeho rozhodnutie za konzistentné. Ak by však účastník urobil nekonzistentné rozhodnutie ako napríklad $C \rightarrow A$ potom je táto trojica označená za kruhovú trojicu (circular triad) a nie je tranzitívna [13].

Nekonzistencia nevyhnutne neznamená, že získané dáta sú chybné. Práve naopak nám táto informácia môže ponúknuť zaujímavé informácie ohľadom výsledku testu. Ak v priemere je väčšina účastníkov nekonzistentná znamená to, že šedotónové videá jednotlivých operátorov sú si podobné a je obtiažne urobiť konzistentné rozhodnutia. V opačnom prípade ak sa nekonzistencia vyskytuje v malej skupinke účastníkov môžeme vyvodiť záver, že títo účastníci nie sú schopní konzistentného rozhodovania a ich skóre by nemuselo byť rátané. Koeficient konzistencie sa definuje rovnicou [13]:

$$\zeta = 1 - \frac{24c}{t^3 - 4t}, \quad (2.38)$$

kde c je počet kruhových trojíc jedného účastníka pre jedno video. Počet kruhových trojíc c sa vypočíta:

$$c = \frac{t}{24}(t^2 - 1) - \frac{1}{2}T, \quad (2.39)$$

kde T sa vypočíta ako:

$$T = \sum_{i=1}^t (p_i - (t-1)/2)^2, \quad (2.40)$$

kde p_i je počet preferencií pre operátor op_i , kde $i = (1, 2, \dots, t)$.

Ak ζ je rovné 1 potom v dátach neexistujú žiadne kruhové trojice. Koeficient ζ sa bude blížiť k 0 ak bude rásť počet kruhových trojíc teda bude rásť nekonzistencia. ζ sa počíta pre každého účastníka zvlášť a pri videu sa udáva priemerná hodnota ζ všetkých účastníkov ktorý hodnotili zadané video [13].

2.6.4 Významnosť

Po vypočítaní koeficientov u a ζ je treba otestovať významnosť (significance). Na otestovanie významnosti koeficientu zhody c otestujeme nulovú hypotézu H_0 ktorá znie: Medzi účastníkmi nie je žiadna zhoda. Nulovú hypotézu môžeme vyvrátiť do hladiny významnosti $\alpha = 0.05$. Na otestovanie významnosti u použijeme chí-kvadrát (chi-squared) test [13]:

$$\chi^2 = \frac{t(t-1)(1 + u(s-1))}{2} \quad (2.41)$$

Na zistenie toho či sú naše získané hodnoty štatisticky významné použijeme pravdepodobnostné tabľky¹ pre χ^2 so stupňami voľnosti (degrees of freedom) vypočítanými ako $t(t-1)/2$ [13].

¹Chi-squared table <https://www.medcalc.org/manual/chi-square-table.php>

2.6.5 Pearsonov korelačný koeficient a koeficient determinácie

Je označovaný písmenom r a je to metrika na vyjadrenie lineárnej korelácie medzi premennými X a Y . Má rozsah od -1 po 1, kde -1 značí úplnú negatívnu lineárnu koreláciu, 0 značí žiadnu lineárnu koreláciu a 1 značí pozitívnu lineárnu koreláciu. Výpočet koeficientu sa dá aplikovať aj na súbor hodnôt. Majme teda súbor hodnôt x_1, \dots, x_n ktorý obsahuje n hodnôt a druhý súbor hodnôt y_1, \dots, y_n ktorý obsahuje n hodnôt potom sa koeficient r vypočíta nasledovne[8]:

$$\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}, \quad (2.42)$$

kde n je veľkosť súboru hodnôt, x_i, y_i sú jednotlivé hodnoty zo súborov a x_i je definované ako priemer súboru $\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i$, analogicky pre y_i . Koeficient determinácie R^2 sa používa pri štatistických modeloch na odhad vzťahu medzi premennými. Vypočíta sa ako druhá mocnina korelačného koeficientu r [12].

Kapitola 3

Návrh

Táto kapitola sa venuje návrhu datasetu videí a návrhu metodiky testovania. Prvá časť rozoberá návrh datasetu aj s ukážkami niekoľkých snímok z videí. V druhej časti sa popisuje návrh metodiky testovania a analyzuje sa pilotná štúdia.

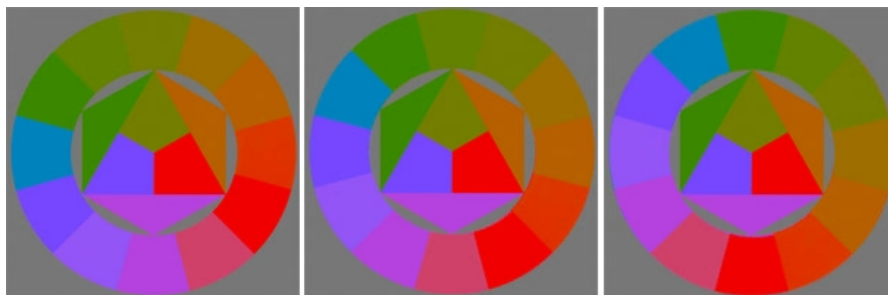
3.1 Dataset

Pri navrhovaní datasetu videí sa muselo dbať na niekoľko dôležitých vecí. Určenie účelu datasetu bolo zrejmé keďže sa práca zaoberá metódami ktoré prevádzajú video do šedotónovej reprezentácie. Zároveň by tieto metódy mali fungovať na obsahovo rôznych videách. Dataset by mal teda pokrývať čo najviac možných typov videa s čo najrozmanitejším obsahom ale zároveň musia byť videá v datasete relevantné. Veľmi dôležitou časťou návrhu je aj konečný počet videí v datasete. Pri príliš malom počte nie sme schopný zahrnúť všetky možnosti ktoré chceme testovať ale na druhej strane je možné skrátiť čas testovania. Pri príliš veľkom datasete môže prísť k tomu, že videá sú si podobné a v podstate sa nimi testuje to isté. Pri získavaní videí do datasetu je taktiež nutné myslieť na to, že videá sú často chránené autorskými právami ktoré neumožňujú ich poskytovanie alebo úpravu.

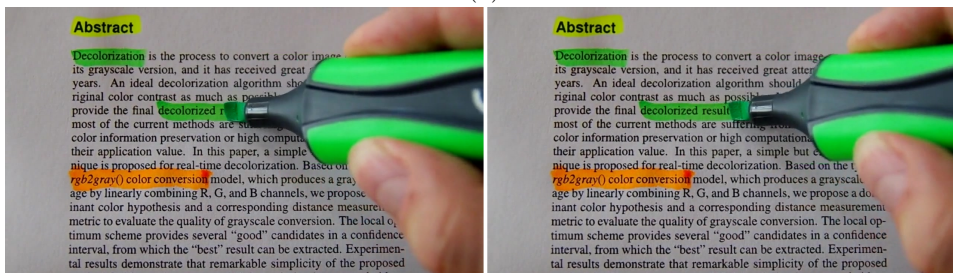
Pri výbere obsahu videí boli možnosti výberu rozdelené do niekoľkých skupín poprípade podskupín. Prvou skupinou boli videá ktoré boli odvodené od datasetu zo štúdie spomínanej v podkapitole 2.4.3. Pre tento typ videí sme sa rozhodli z dôvodu, aby exitovala možnosť priameho porovnania medzi týmito datasetmi. Takto by bolo možné porovnať výkon metódy pri jedinom obrázku a zároveň na jemu podobnému videu. Videá boli vytvorené dvomi spôsobmi a to vytvorením videa úpravou obrázkov z datasetu (obr. 3.1a) alebo natočením vlastného videa ktoré bolo inšpirované datasetom (obr. 3.1b).

Ďalšou skupinou boli videá ktoré sú odvodené od filmovej produkcie. Bolo dôležité považovať aké rôzne druhy filmov začleniť do datasetu, či už sa jednalo o klasický film alebo o rôzne typy animácií ako napríklad počítačové, stop-motion, kreslenú (obr. 3.2b). Dynamika scény bola tiež považovaná za dôležité kritérium pri výbere, pretože akčná scéna (obr. 3.2a) sa od konverzačnej scény môže líšiť väčším počtom strihov, trasením kamery alebo inými výraznými obrazovými prechodmi. Pri tejto skupine sú veľmi dôležité licencie ktorými je video chránené. Z dôvodu, že veľké filmové štúdiá sú veľmi striktné ohľadom tejto záležitosti bola zvolená možnosť použiť filmové diela s voľnou licenciou konkrétne z Blender foundation¹. Použitie filmov s touto licenciou umožňuje bezproblémové zdieľanie datasetu.

¹<https://www.blender.org/>

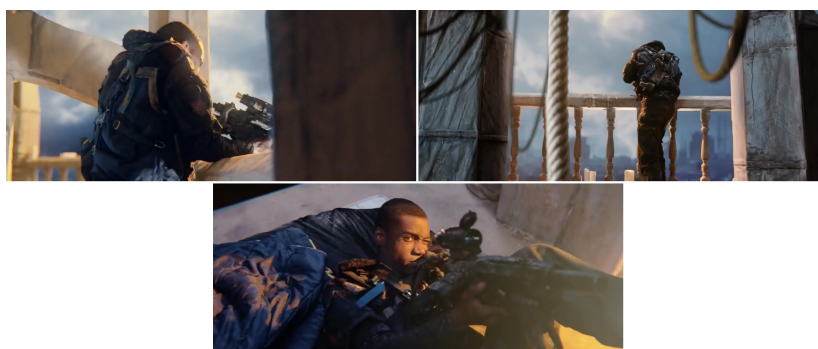


(a)

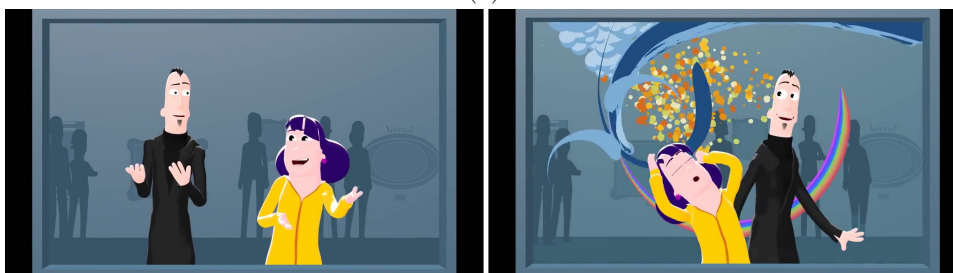


(b)

Obr. 3.1: Ukážka pár snímkov videí inšpirovaných datasetom z kapitoly 2.4.3.



(a)



(b)

Obr. 3.2: Ukážka (a) akčnej scény a (b) animácie.

Veľkú skupinu tvoria videá zobrazujúce krajinu, stromy, kvety a zvieratá. Pri týchto videách môžu mať ľudia nejakú predstavu ako by odfarbené video malo vyzeráť. Vyplýva to z toho, že už videli niekoľko čiernobielych filmov a fotografií. Zároveň sa v prírode vyskytuje veľmi veľa farebých kombinácií a vzorov ktoré poslúžia na otestovanie metód. V tejto skupine bolo použité video dostupné na YouTube, pričom jeho autor dal súhlas k úprave videa a k použitiu v tomto datasete. Jedná sa o zábery lesa ktoré sú zhotovené pomocou dronu (obr. 3.3a). Video bolo zaradené do datasetu aj na základe rastúcej popularity dronových záberov.

Špecifickou kategóriou je zobrazenie ľudí (obr. 3.3b). Opäť ako pri predchádzajúcej skupine majú ľudia určité očakávanie ako vyzerá ľudská pokožka po odfarbení. Preto je táto skupina vhodná na otestovanie toho či metódy dokážu previesť video človeka v nejakom šedotónovom odtieni ktorý je ľuďmi považovaný za prijateľný a ktorý nepôsobí rušivo.

Ďalšie skupiny zahŕňajú videá ako napríklad scény renderované počítačom (obr. 3.3c), scény kde je farebná paleta pomerne malá (obr. 3.3d), scény z veľkou farebnou paletou alebo videá zamerané na rôzne materiály 3.3b.

Videá nie sú striktné zaradené len v jednej skupine ale môžu mať vlastnosti ktoré by mohli byť zaradené do viacerých skupín. Takéto videá umožnia otestovať metódu z viacerých ohľadov.

Dĺžka jednotlivých videí bola veľmi dôležitá téma. Pri príliš dlhých videách by dochádzalo k opadnutiu záujmu od účastníkov testu, čo by pravdepodobne znamenalo nedopozorovania videa dokonca alebo stratu motivácie na dokončenie testu. Tiež by to viedlo k nárastu trvania testu. Naopak pri príliš krátkych videách by nebolo možné otestovať niektoré správanie metód ako napríklad pri videu ktoré obsahuje množstvo strihov a podobne. Najskôr boli videá koncipované ako 30 sekundové sekvencie, no po zvážení vyššie vymenovaných nevýhod bola dĺžka stanovená na približne 10 sekúnd. Priemerná dĺžka videa v datasete je zhruba 9 sekúnd, pričom najdlhšie video má 13 sekúnd a najkratšie 4 sekundy.

Väčšina videí je v HD rozlíšení okrem videí ktoré vychádzajú z datasetu z kapitoly 2.4.3. Rozlíšenie v HD bolo zvolené z dôvodu, že dnes sa už jedná o štandardné rozlíšenie ktoré poskytuje dostatočnú kvalitu. Videá ako *dominika*, *text* a *fullHouse* boli zachytené pomocou digitálneho fotoaparátu Pentax K-50. Tieto videá sú takisto v HD rozlíšení.

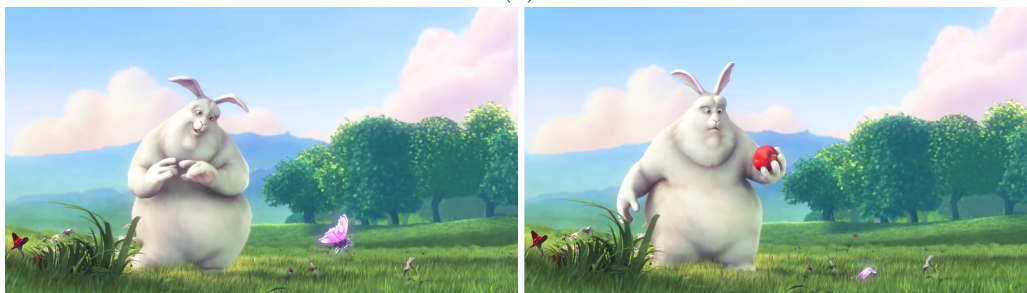
Pre celý dataset bolo navrhnutých 37 videí. Nakoniec bol tento počet po úvahe zredukovaný na 24 a vznikol tak zredukovaný dataset. Do tohto datasetu sa nedostali videá ktoré boli istým spôsobom duplicitné alebo nie celkom vyhovovali rozdeleniu videí po koncepcnej stránke ktoré je bližšie rozobraté v kapitole 3.2. Všetky videá boli pred zaradením do zredukovaného datasetu prevedené do šedotónu len za pomoci jasovej zložky (2.1.2), aby sa zistilo či sa u nich nevyskytuje blikanie. Tento artefakt sa objavil u videí *trees* a *girls*. Dataset bude zverejnený so všetkými videami aby bola zarúčená možnosť výberu používateľmi. Všetky videá sú chované licenciou CC alebo na ich zdieľanie bol získaný súhlas autora. Táto licencia umožňuje zdieľanie videí a je v pláne dataset zverejiť aby bol dostupný k použitiu pri testovaní nových metód. Odkazy na všetky dostupné videa na internete sú v prílohe B.



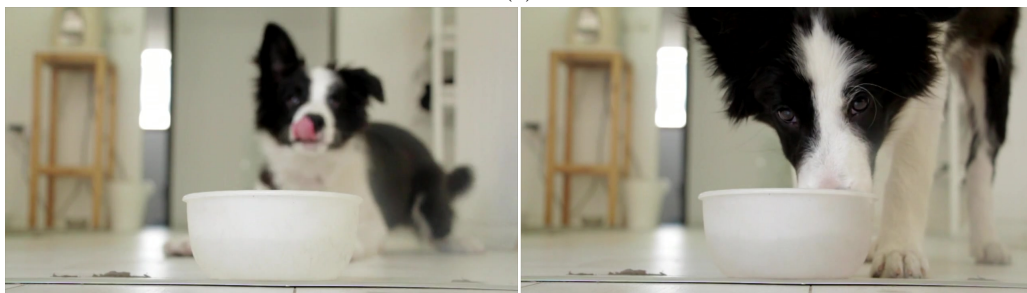
(a)



(b)



(c)



(d)

Obr. 3.3: Ukážka videí z datasetu reprezentujúce (a) počítačový render, (b) prírodu, (c) video s malou farebnou paletou, (d) video s človekom v zábere a rôznymi materiálmi (látkami)

3.2 Metodika testovania

Pri návrhu metodiky sme sa riadili informáciami poskytnutými v kapitolách 2.2 a 2.4. Rozhodnutie koncipovať test ako rozhodovaciu úlohu (2.2.2) bolo pomerne priamočiare, pretože niektoré metódy neposkytujú voľbu parametrov alebo sú vo väčšine prípadov výhradne testované z východiskovými hodnotami parametrov. Taktiež sme chceli porovnávať metódy medzi sebou a nie aby používateľ nastavoval výsledky metód k určitej referencii. Z rozhodovacích metód boli na výber dve možnosti testovania a to rating scale alebo two-alternative forced choice (2AFC), metóda yes-no nie je príliš vhodná v tejto situácii. V štúdiu 2.4.1 bola na testovanie použitá možnosť rating scale no pri dlhšom zvážení bolo rozhodnuté túto metódu nepoužiť z viacerých dôvodov. Prvým dôvodom bolo, že rating scale je menej presná ako 2AFC. Druhým dôvodom bolo, že vďaka popularite štúdie 2.4.3 by bolo zaujímavé nadviazať na ňu otestovaním videí v podobnom štýle.

Do testu bolo vybraných šesť metód (tab. 3.1). Z týchto metód boli tri časovo závislé metódy, dve metódy časovo nezávislé a metóda využívajúca len jasovú zložku Y 2.1.2 ktorá posluží ako referenčné šedotónové video. Metóda Decolorize07 bola vybraná z dôvodu, že v štúdiu popísanej v kapitole 2.4.3 bola najúspešnejšia a bude zaujímavé sledovať jej výkonnosť medzi časovo závislými metódami. Použité metódy Lee10 [16] bolo zvolené z dôvodu, že zachováva chromatický kontrast a zároveň udržiava jasovú konzistenciu.

Metóda	Ref	Z/N	Implementácia	Parametre
Hu14	[5]	Z	vlastná C++	$\lambda=0.9, \tau=15, \delta=10, \gamma=1.6, \theta = 0.02$
Tao17	[17]	Z	vlastná C++	$\beta=0.5$
Kim09	[9]	Z	vlastná C++, časť P.Pospíšil	$\alpha=1, \beta=0.5$
Decolorize07	[10]	N	Matlab	effect=0.5, scale=25, noise=0.001
Lee10	[16]	N	M. Molek C++	k=3, $\alpha=0.5$
CIE Y		N	vlastná C++	–

Tabuľka 3.1: Prehľad metód vybraných v návrhu, N značí časovo nezávislú metódu, Z je označenie časovo závislej metódy.

Boli navrhnuté dva typy testov: Test bez referencie (preferenčný), Test s referenciou zameraný na presnosť prevodu.

Pri teste bez referencie boli účastníkom experimentu zobrazené dve šedotónové videá vedľa seba (obr. 3.4b) a mali sa rozhodnúť pre to ktoré preferujú. Presné zadanie úlohy bolo nasledujúce: Z dvojíc videí vyberte to, ktoré je Vám vizuálne príťažlivejšie.

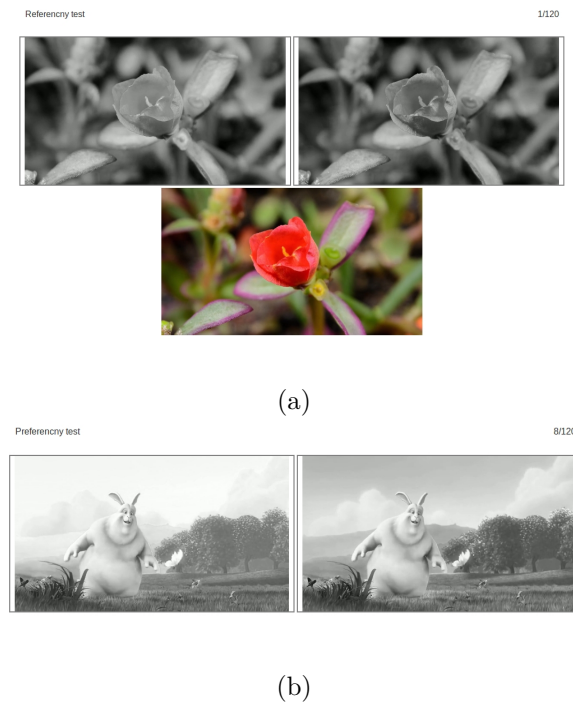
Test s referenciou zobrazil účastníkom experimentu dve šedotónové videá pod ktorými sa nachádzalo originálne farebné video ako referencia (obr. 3.4a). Úlohou bolo vybrať video ktoré sa viac podobá na originálne video. Konkrétne zadanie znelo: Z dvojíc videí vyberte to, ktoré podľa Vás najlepšie reprezentuje farby z pôvodného farebného videa.

Po teste nasleduje krátky dotazník kde sa v prvej časti dotazníku pýtame na pohlavie, vek a či daný účastník netrpí farbosleposťou alebo porušením farbobitu. V druhej časti dotazníka zisťujeme skúsenosti so spracovaním obrazu jednotlivých účastníkov. Na výber mali jednu z troch možností a to: žiadne, amatérske alebo profesionálne. V ďalšej otázke sa snažíme zistiť čo účastníkom najviac vo videách prekážalo alebo ich rozptýľovalo. Keďže očakávame, že väčšina účastíkov nebude mať profesionálne skúsenosti z tejto oblasti

a nevedeli by správne pomenovať rozptýľujúce faktory tak im bolo poskytnutých zopár možností. Pri výbere mali vybrať dve z nasledovných možností: blikanie videa, nerozlišiteľné farby (chýbajúci kontrast), neprirodené šedotónové farby, strata detailov, zhoršenie kvality (štvorčekovanie). Táto časť dotazníku bola inšpirovaná štúdiou z podkapitoly 2.4.2.

Použitím metódy 2AFC porovnávame šedotónové videá štýlom každý s každým pre všetky videá z datasetu. Pre porovnanie bolo použité webové riešenie z bakalárskej práce od p. Ševčíka [21]. Počet porovnaní pre jedného účastníka testu sa vypočíta ako $t(t-1)/2$, kde t je počet metód 2.4.3, v našom prípade teda $6 \times 5/2 = 15$ porovnaní. Tetno počet porovnaní je pre jedno video ale keďže v datasete je 24 videí tak konečný počet porovnaní je $15 \times 24 = 360$ porovnaní pre účastníka. Takéto množstvo porovnaní je príliš veľké pre jedného človeka. Ak by sme ráтали s tým, že si používateľ pozrie každé video do konca bez času na rozmyslenie tak by sa v priemere jednalo o 54 minút (pri priemernej dĺžke videa 9 sekúnd). Po takto dlhú dobu nie je pre človeka možné udržať koncentráciu na zadanú úlohu. Z tohto dôvodu bola vykonaná pilotná štúdia aby sa zistil primeraný počet porovnaní a aby boli zapracované pripomienky používateľov.

V pilotnej štúdii bol zvolený počet videí na 8, čo znamenalo 120 porovnaní na jedného človeka. S týmto počtom porovnaní trval experiment zhruba 25 minút. Dataset bol teda rozdelený na 3 skupiny po 8 videí pre každý test, čo je dokopy 6 variatov ktoré mohol účastník dostať na otestovanie. Videá boli do týchto skupín rozdelené tak aby v každej skupine bolo čo najväčšie pokrytie rozdelenia videí v 3.1. Štúdie sa zúčastnilo 10 ľudí. Väčšina účastníkov považovala dĺžku testu za primeranú, no niektorými bola označená za hraničnú. Z tohoto dôvodu nebol počet porovnaní navýšený. Ďalšou pripomienkou bolo, že videá sa neopakujú a pre účastníkov bolo náročné sledovať toľko videí naraz. Táto pripomienka bola akceptovaná a bolo dopracované opakovanie videí. V dvoch prípadoch sa vyskytla pripomienka, že videá sa pomaly načítajú. Tejto pripomienke nebolo vyhovené z dôvodu zníženia kvality videí kvôli menšej veľkosti čo by mohlo ovplyvniť konečný výsledok.



Obr. 3.4: Ukážka referenčného (a) a preferenčného testu (b) webového rozhrania.

Kapitola 4

Implementácia

Kapitola sa venuje samotnej implementácii jednotlivých častí práce. Na začiatku kapitoly je popis systému TMS do ktorého boli všetky časti práce implementované. Potom nasleduje popis implementácie knižnice pre načítanie videa v systéme TMS. Po tejto podkapitole nasleduje popis implementácie jednotlivých metód popísaných v 2.3.

4.1 Tone mapping studio

Jedná sa o framework, označovaný ako TMS, ktorý je vyvíjaný od roku 2004 pod vedením doc. Čadíka. Je určený na implementáciu metód (operátorov) ktoré fungujú na princípe tone mappingu alebo prevodu obrázkov do šedotónu. V kontexte TMS sa metóda na tone mapping alebo na prevod do šedotónu nazýva operátor a termín metóda je používaný v kontexte OOP. TMS obsahuje rozhranie na definíciu nových operátorov, ďalej sa stará o načítanie súborov v rôznych formátoch a ich ukladanie do formátu TIFF a EXR. Podporuje taktiež rôzne farebné modely a prevody medzi nimi.

TMS poskytuje dva varianty použitia a to buď konzolového rozhrania ktoré sa spustí pomocou súboru `tmocmd` alebo grafického rozhrania spustiteľného súborom `tmogui`. Funkcionalitou sú si tieto možnosti ekvivaentné. Pri práci na tomto projekte bolo využívané len konzolové rozhranie. TMS má jednoduchú schému použitia a to takú, že používateľ si zvolí vstupný súbor, operátor ktorý chce použiť, nastaví jeho parametre a následne sa aplikuje operátor na vstupný súbor a výsledok sa uloží do výstupného súboru.

Dôležitou triedou je trieda TMO. V rámci frameworku TMS všetky operátory dedia z tejto triedy. Trieda TMO obsahuje množstvo metód z ktorých je najdôležitejšia metóda `virtual int Transform()`, pretože implementuje samotné chovanie operátora.

Trieda `TMOImage` poskytuje rozhranie pre prácu s obrázkom. Medzi najpoužívanejšie a najdôležitejšie metódy z pohľadu používateľa patria nasledovné: `virtual double* GetData()` pre získanie ukazateľa na obrazové dáta, `virtual int GetWidth()` a `GetHeight()` pre získanie šírky a výšky obrázku, `virtual int Convert(int format, bool fast=false)` pre prevod obrázku do iného farebného modelu. Takisto sú v triede obsiahnuté metódy na otvorenie obrázku `virtual int Open(const char *filename)` a metódu `virtual int SaveWithSuffix(const char *suffix, int fileFormat=TMO_TIFF_8)` na uloženie obrázku ktorého názov obsahuje aj názov použitého operátora. Na generovanie Makefile súborov využíva TMS nástroj Cmake¹.

¹<https://cmake.org/>

4.2 TMOvideo a TMOv

TMS nedokázalo pracovať s videami takže bolo nutné túto časť doimplementovať. Bola na implementovaná trieda TMOv ktorá dedí od triedy TMO aby boli dostupné všetky metódy. Zároveň pridáva zopár nových: `virtual int SetVideos(TMOVideo& src, TMOVideo& dst)` metóda ktorá nastaví parametre vstupného a výstupného videa, `virtual int SetOutVideo(TMOVideo& dst)` ktorá nastavuje parametre len pre výstupné video. Najdôležitejšou metódou triedy TMOv je `virtual int TransformVideo()=0` v ktorá implementuje samotné chovanie operátora podobne ako metóda pre obrázky popísaná v kapitole 4.1.

Ďalšou implementovanou triedou je trieda TMOVideo ktorá poskytuje rozhranie pre prácu z videom. Na prácu z videom bola použitá knižnica OpenCV. Keďže OpenCV načítava jednotlivé snímky videa do typu `cv::Mat` bolo nutné naimplementovať metódy ktoré sa starajú o prevod typu `cv::Mat` na TMOImage a naopak. Konkrétne sa jedná o metódy `virtual int cvMatToTMOImage(TMOImage &img, cv::VideoCapture c, cv::Mat &frame)` a `virtual int TMOImageToCvMat(TMOImage &img, cv::Mat &frame)`. Snímok obrazu sa dá získať dvoma metódami v závislosti od toho či chce používateľ snímok typu `cv::Mat` použije metódu `virtual int GetMatVideoFrame(cv::VideoCapture c, int frameNumber, cv::Mat &frame)` alebo keď potrebuje snímok ako TMOImage tak môže použiť metódu `virtual int getTMOImageVideoFrame(cv::VideoCapture c, int frameNumber, TMOImage &img)`. Pre uloženie snímku do videa existujú taktiež dve varianty `virtual int setMatFrame(cv::VideoWriter out, cv::Mat frame)` pre typ `cv::Mat` alebo pre typ TMOImage je to metóda `virtual int setTMOImageFrame(cv::VideoWriter out, TMOImage &img)`. Ďalšími dôležitými metódami sú: metódy na získanie rozmerov `virtual int GetHeight()` a `virtual int GetWidth()` a takisto metódy na získanie objektov ktoré sú potrebné pre čítanie a zápis snímok do videa, teda metódy `virtual cv::VideoCapture getVideoCaptureObject()` resp. `virtual cv::VideoWriter getVideoWriterObject()`.

V triede TMOCmd bola pridaná funkcionálna aby mohli byť načítané aj videá a bola doplnená funkcionálna o prekonvertovanie obrázkov v zadanom adresári. Ako prvé sa zistí, či zadaná cesta k vstupnému súboru je cesta k súboru alebo k adresáru. Najskôr sa použije funkcia `stat()` ktorá naplní štruktúru `stat` informáciami o súbore alebo adresári ktorý bol zadaný ako parameter funkcii `stat()`. Informácia o tom či sa jedná o súbor alebo adresár je uložená v štruktúre `stat` v premennej `st_mode`. Potom sa pomocou makra `S_ISREG(stat.st_mode)` a `S_ISDIR(stat.st_mode)` zisťuje či sa jedná o súbor resp. adresár.

Ak sa jedná o súbor zistí sa či jeho prípona odpovedá video formátom mp4, wmv alebo avi. Keď je súbor identifikovaný ako video bude vykonané dynamické pretypovanie z triedy TMO na TMOv ak sa toto pretypovanie podarí vieme, že je implementovaná metóda `TransformVideo()` ktorá je následne invokovaná. Pri zlyhaní pretypovania vieme, že operátor nemá implementovanú metódu `TransformVideo()`. V tomto prípade sa bude pre jednotlivé snímky videa v cykle invokovať metóda `Transform` a jednotlivé snímky po aplikácii operátora budú uložené opäť ako video. Ak je súbor identifikovaný ako obrázok tak funkcionálna je zachovaná tak ako bola.

Ak sa jedná o adresár je tento adresár prehľadaný a je vytvorený vektor názvov všetkých nájdených obrázkov. Následne je na každý obrázok z tohto zoznamu aplikovaný zadaný operátor.

4.3 Real-time Decolorization using Dominant Colors

Pri tomto operátore bol implementovaný prevod obrázka ako aj videa. Pri implementácii bola použitá knižnica OpenCV. V hlavičkovom súbore `TM0Hu14.h` boli v štruktúrach `lessVec3b` a `lessVec4d` nadefinované operátory pre porovnávanie 3-D resp. 4-D vektorov. Ďalej boli naimplementované nasledujúce metódy:

- `cv::Mat getEdgeMat(cv::Mat)` - metóda aplikuje Canny hranový detektor pomocou OpenCV funkcie `cv::Canny` a získa hrany zo vstupného obrazu.
- `kmeansColorQuantization(const cv::Mat3b&, cv::Mat3b&)` - Pomocou funkcie `cv::kmeans` zredukujeme počet farieb v obraze na 256
- `getPalette(std::map&, cv::Mat&)` - Metóda získa jednotlivé farby z obrázku vo farebnom modeli Luv a ich percentuálne zastúpenie.
- `std::map getDominantColorDescriptor(std::map)` - Získajú sa dominantné farby vid' 2.3.1 v Luv modeli s ich percentuálnym zastúpením.
- `cv::Vec3d rgb2Luv(cv::Vec3i)`, `cv::Vec3d Luv2rgb(cv::Vec3d)`, `cv::Vec3d xyz2bgr(cv::Vec3d)` - Prevody farebných priestorov pre jednu farbu reprezentovanú vektorom.
- `std::vector getBestWeightsCandidate(std::map, cv::Mat, cv::Mat, cv::Mat)` - Zoradí všetky kombinácie váh od najlepšej po najhoršiu podľa metriky ξ (2.2).
- `cv::Mat getHistogram(cv::Mat)` - Získa histogram pre snímok v Luv.
- `std::map getGrayscalePalette(float, float, float, std::map)` - Metóda prevedie vektor dominantných farieb do šedotónovej reprezentácie na základe váh.
- `double getXiMetric(std::map)` - Výpočet metriky ξ podľa vzťahu 2.2.
- `double getHMetric(cv::Vec4d, cv::Vec4d)` - Metóda pre výpočet metriky H podľa 2.5.
- `double getChangingRate(double)` - Metóda vypočíta faktor zmeny.
- `cv::Vec4d getClosestWeight(cv::Vec4d, std::vector, int)` - Návratová hodnota metódy je najbližšia váha podľa vzťahu 2.7.
- `int Transform()` - Hlavná metóda pre dekolizáciu obrázkov. Na začiatku je každý farebný kanál pridelený do vlastnej matice typu `cv::Mat`. Následne sú tieto matice podvzorkované o polovicu pomocou funkcie `cv::resize`. Potom sú získané hranové obrazy pre každý kanál ktoré sú vynásobené maticami farebných kanálov a sú spočítané aby sme dostali maticu obsahujúcu I-edge (2.3.1). Ďalej sa nakvantizujú farby a získajú sa dominantné farby. Ako ďalšie sa do premennej `luvBgrPalette` ktorá je typu `std::map<cv::Vec4d, cv::Vec3d, lessVec4d>` vloží ako prvý vektor Luv farba s jej percentuálnym zastúpením v obraze a ako druhý vektor BGR farba. Po zoradení najlepších váh sa na obrázok aplikuje najlepšia z nich.

- `int TransformVideo` - Hlavná metóda pre dekolORIZáciu videa. Metóda prebieha v cykle až do načítania posledného snímku videa. Začiatok metódy je identický s metódou `Transform()` až po získanie zoradených váh. Následne sa získa histogram aktuálneho snímku ak sa jedná o prvý snímok videa tak je histogram a váhy priradené do predchádzajúceho snímku a načíta sa ďalší snímok ktorý je teraz aktuálny. Ak sa nejedná o prvý snímok z videa potom sú histogrami aktuálneho snímku a predchádzajúceho porovnané pomocou OpenCV funkcie `cv::compareHist` a je vypočítaný faktor zmeny. Ak je rozdiel histogramov väčší ako θ potom sú váhy upravené v zmysle rovnice 2.7. Výsledné váhy sú použité na dekolORIZáciu aktuálneho snímku.

4.4 Robust Color-to-gray via Nonlinear Global Mapping

Implementácia tejto metódy pre prevod obrázkov bola implementovaná pánom Pospíšilom[14], lenže metóda obsahuje aj temporálnu časť pre videá ktorá bola doimplementovaná v tejto práci. Pri implementácii metódy `int TMOVideo()` boli využité kódy metód od p. Pospíšila, pretože pri temporálnej časti sa využívajú aj časti z priestorového riešenia viď podkapitolu 2.3.3. Metóda `int TMOVideo()` pracuje v cykle až dokým nie je načítaný posledný snímok zo vstupného videa. Snímok videa je načítaný pomocou metódy `getTMOImageVideoFrame` z knižnice `TMOVideo` popísanej v podkapitole 4.2. Popri počítaní vektorov u a v je počítaný aj vektor t popísaný v rovnici 2.21. Potom sa vypočíta matica M_s a vektor b_s . V prípade, že sa jedná o prvý snímok z videa tak je snímok dekolORIZovaný ako pri obrázkoch a po zápise dekolORIZovaného snímku je jeho hodnota priradená do premennej `previousGrayFrame` ako predchádzajúci šedotónový snímok a hodnota farebného snímku sa zapíše do premennej `previousFrame` ako predchádzajúci snímok a následne sa v ďalšej iterácii načíta snímok ktorý je teraz aktuálnym snímkom. Ak sa nejedná o prvý snímok tak sa vypočítajú hodnoty M_t a b_t viď rovnice 2.34 a 2.35. Následne sa vypočítajú M_v , b_t a \hat{x}_{video} a dekolORIZuje sa snímok. Pred pred zápisom snímku sa nastaví dáta snímku pomocou `TMOImage` metódy `SetData` a potom sa pomocou `TMOVideo` metódy `setTMOImageFrame` zapíše. Opäť sa snímky popriradzujú do premenných ako bolo popísane skôr.

4.5 Video decolorization Using visual proximity coherence optimization

Implementácia metódy `int Transform()` pre obrázky zahŕňala naimplementovanie rovnice 2.11. Na jej minimalizáciu boli použité knižnice `dlib` a `pallas solver` ale výsledky nezodpovedali výsledkom ktoré sú uvedené v článku. Minimalizácia bola potom vypočítaná spôsobom ktorý je prezentovaný v článku [4], kde výsledky už boli reprezentatívne.

V metóde `std::vector getPixelDifferences(std::vector, int)` sa počíta rozdiel medzi jednotlivými snímkami 2.3.2. Na výpočet optického toku ktorý sa používa v rovniciach 2.8 a 2.9 je implementovaná metóda `virtual int getOpticalFlow(cv::Mat, cv::Mat, cv::Mat&)` ktorá využíva OpenCV funkciu `cv::calcOpticalFlowFarneback` na zistenie optického toku. Algoritmus HPD využívajúci vzťah 2.8 je implementovaný metódou `virtual int getDifferentialRefinementFrame(cv::Mat, cv::Mat, cv::Mat&)`. Algoritmus MPD na výpočet coherence refinement frame podľa vzťahu 2.10 implementuje metóda `virtual int getCoherenceRefinementFrame(cv::Mat, cv::Mat, cv::Mat, cv::Mat&)`. Výsledok implementácie nekorešponduje s výsledkami prezentovanými v článku. Dôvodom je nedostatočne vysvetlený spôsob ako sa k výslednému coherence refinement frame

dostať. Algoritmus pre LPD je naimplementovaný v metóde `virtual int getLowProximityFrame(cv::Mat, cv::Mat, cv::Mat, cv::Mat& výsledky tejto metódy nie sú správne pretože zrejme došlo k chybe keď neboli energie v rovnici 2.3.2 minimalizované spoločne ale oddelene a je možné, že minimalizácia použitá v metóde int Transform() nefunguje práve najlepšie pre tento operátor. Pri implementácii klasifikátora sa takisto vyskytli problémy, kedy neboli uvedené hodnoty niektorých premenných a nie celkové pochopenie napísaného algoritmu čo viedlo k nefunkčnosti klasifikátora. Nefunkčnosť niektorých metód a hlavne klasifikátora vďaka programátorským chybám a neuvedeným skutočnostiam v článku vedie k tomu, že tento operátor neposkytuje reprezentatívne výsledky.`

4.6 Použité knižnice

4.6.1 OpenCV

Jedná sa o veľmi známu open source knižnicu² zameranú na počítačové videnie a strojové učenie. OpenCV má veľkú komunitu používateľov. Knižnica obsahuje viac ako 2500 optimalizovaných algoritmov ako napríklad detekcia tváre, rozpoznávanie objektov a mnoho ďalších. Takisto podporuje maticové operácie, prevody medzi farebnými modelmi a mnoho iných funkcií.

4.6.2 matrix.h

Slúži pre prácu s maticami a ako už z názvu vyplýva jedná sa o hlavičkovú knižnicu³. Obsahuje triedy pre ukladanie vektorov a matíc. Umožňuje základné matematické operácie ale taktiež maticové operácie typu pseudoinverzia a podobne.

4.6.3 Dlib

Táto knižnica⁴ je open source a zameriava sa na strojové učenie, optimalizáciu, spracovanie obrazu a iné typy výpočtov. Jedná sa o platformovo nezávislú knižnicu ktorá si zakladá na ľahkej použiteľnosti.

4.6.4 Pallas Solver

Knižnica⁵ sa zameriava na globálnu optimalizáciu funkcie. V momentálnom štádiu sú implementované štyri algoritmy pre globálnu optimalizáciu. Konkrétne sa jedná o algoritmy: Basin hopping, Brute, Differential Evolution a Simulated Annealing.

²<https://opencv.org/>

³<http://www.rejtrix.net/>

⁴<http://dlib.net/>

⁵<https://github.com/latture/pallas-solver>

Kapitola 5

Testovanie

5.1 Základné údaje

Testovania sa zúčastnilo 60 ľudí ktorý vykonali 7200 párových porovnaní. Na teste ktorý bol zameraný na presnosť (s referenciou) sa zúčastnilo 30 ľudí a počet účastníkov v druhom teste zameranom na preferenciu (bez referencie) bol takisto 30. Dáta o užívateľoch sa ukladali do preferenčnej matice (2.6.1) o rozmeroch 6×6 . Na prevod do štandardného z-score bol použitý zákon porovnávacích rozhodnutí konkrétne prípad V, pomocou kódu¹ pre Matlab.

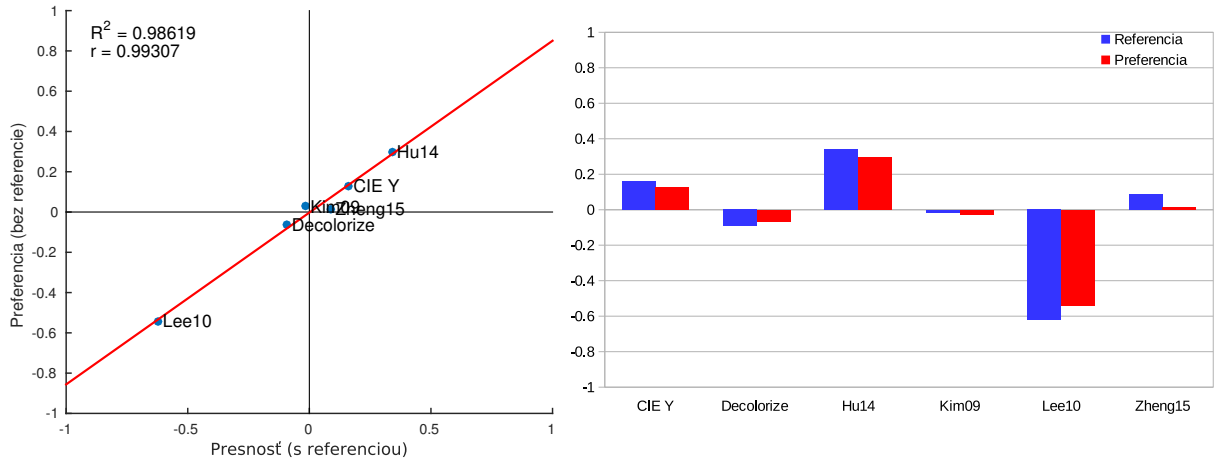
Oproti návrhu (3.2) nebude použitá metóda Tao17 kvôli dôvodom popísaným v 4.5. Táto metóda bola nahradená časovo nezávislou metódou Zheng15 [19] aby bol zachovaný rovnaký počet metód. Metóda bola spúšťaná s parametrami: $\alpha = 0.2, \beta = 0.2, \gamma = 0.5, \theta = 0.0$. Metóda bola naimplementovaná p. Pospíšilom [14].

5.2 Výsledky pre celý dataset

Pre porovnanie boli vytvorené preferenčné matice pre zvlášť pre presnosť a pre preferenciu. Z obr. 5.1 môžeme vidieť, že metóda Hu14 je najlepšia v obidvoch experimentoch, zatiaľ čo metóda Lee10 je najhoršia v obidvoch experimentoch. Metóda Kim09 ako druhá časovo závislá metóda bola predstihnutá metódami CIE Y a Zheng15 čo by sa dalo považovať sa prekvapivý výsledok. Metóda Decolorize ktorá bola v štúdiu popísanej v 2.4.3 jedna z najlepších je teraz na chvoste hodnotenia. Takisto za prekvapivý výsledok by sa dalo označiť druhé miesto referenčnej metódy CIE Y v obidvoch experimentoch. Mohlo by to byť tým, že táto metóda produkuje také výsledky na aké sú ľudia zvyknutý a tým pádom sú im jej výsledky sympatickejšie. Je zaujímavé, že všetky metódy okrem Kim09 dosiahli lepšie výsledky v teste zameranom na presnosť s referenciou, čo by mohlo indikovať nejaký malý vplyv referencie na výsledok.

Porovnaním presnosti a preferencie si môžeme všimnúť isté trendy. Hodnoty Pearsonovho korelačného koeficientu $r = 0.99$ a koeficientu determinácie $R^2 = 0.98$ naznačujú, že existuje v celku silná korelácia medzi prevodom do šedotónu s referenciou a bez referencie. Pri pohľade na výsledky metód CIE Y, Decolorize, Hu14, Kim09 si môžeme všimnúť, že výsledky týchto metód sa zmenili len veľmi málo v porovnaní medzi experimentami.

¹Law of comparative judgement <http://www.dtic.mil/dtic/tr/fulltext/u2/a543806.pdf>



Obr. 5.1: Celkové výsledky obidvoch testov. Vľavo je porovnanie experimentov pre presnosť a preferenciu. Vpravo je celkové skóre pre obidva experimenty.

5.3 Výsledky pre jednotlivé videá

Ďalším krokom bolo analyzovanie dát pre jednotlivé videá (tab. 5.1). Pre každé video bolo z-score vypočítané individuálne pre presnosť ako aj pre preferenciu. Grafy porovnania experimentov sú v prílohe A. Pre jednotlivé videá boli takisto vypočítané koeficienty r a R^2 vo väčšine prípadov dosahovali vysokých hodnôt (viď prílohu A grafy oilColors, spiral, landscapeSajfa a iné), čo by mohlo znamenať, že existuje korelácia medzi prevodom do šedotónu s referenciou a bez referencie. V tabuľke 5.1 uvádzame koeficient zhody u , priemerný koeficient konzistentnosti ζ a takisto hodnoty χ^2 testu s 15 stupňami voľnosti (2.6.4) spolu so získanými p – hodnotami. Podrobný popis týchto koeficientov je v kapitole 2.6. Výsledky χ^2 testu poukazujú vo väčšine prípadov, že medzi používateľmi je nejaká zhoda, pretože ich p – hodnoty sú pod pravdepodobnostnou hladinou $\alpha = 0.05$. Pri týchto videách sa teda podarilo vyvrátiť nulovú hypotézu H_0 2.6.4 ktorá znie: Nie je žiadna zhoda medzi účastníkmi. Prípady kedy sa hypotézu nepodarilo vyvrátiť sú popísané v podkapitole 5.3.1. Hodnota ζ dosahuje väčšinou vyšších hodnôt čo naznačuje konzistentnosť účastníkov no pri koeficiente u sú hodnoty skôr menšie čo znamená, že medzi účastníkmi bola pomerne malá zhoda. Výnimočnejšie prípady obidvoch koeficientov sú popísané v podkapitole 5.3.1.

	u	ζ (priem.)	χ^2	p (15 st.v.)	Poradie podľa skóre					
25_vid (pres.)	0.363	0.763	64	<0.001	L	D	H	K	Y	Z
25_vid (pref.)	0.396	0.938	68.4	<0.001	L	D	H	Z	K	Y
action (pres.)	0.028	0.475	18.8	>0.2	K	H	Y	D	Z	L
action (pref.)	0.206	0.688	42.8	<0.001	Y	D	K	H	Z	L
balls (pres.)	0.164	0.675	37.2	<0.002	Y	Z	H	K	L	D
balls (pref.)	-0.04	0.7	9.6	>0.2	K	Y	H	L	Z	D
dog (pres.)	-0.0014	0.388	14.8	>0.2	H	D	K	Y	L	Z
dog (pref.)	0.04	0.475	20.4	<0.2	D	H	Z	K	L	Y
portrait (pres.)	0.209	0.75	43.2	<0.001	H	Y	K	Z	D	L
portrait (pref.)	0.321	0.738	58.4	<0.001	H	K	Y	Z	D	L
red (pres.)	0.576	0.85	92.8	<0.001	Y	H	K	Z	D	L
red (pref.)	0.221	0.8	44.8	<0.001	H	Y	K	Z	D	L
scand. (pres.)	0.351	0.7	62.4	<0.001	H	Z	Y	L	D	K
scand. (pref.)	0.167	0.725	37.6	<0.002	Z	H	Y	L	D	K
tetris (pres.)	0.123	0.713	31.6	<0.01	Y	Z	H	K	D	L
tetris (pref.)	0.239	0.775	47.2	<0.001	Y	Z	H	D	K	L
bunny (pres.)	0.143	0.775	34.4	<0.005	D	H	Y	Z	L	K
bunny (pref.)	0.052	0.688	22	<0.2	H	Z	D	Y	L	K
colorWh. (pres.)	0.333	0.725	60	<0.001	Z	D	H	K	L	Y
colorWh. (pref.)	0.224	0.925	45.2	<0.001	K	D	H	Z	L	Y
cooking (pres.)	0.158	0.613	36.4	<0.002	Z	H	D	K	Y	L
cooking (pref.)	0.259	0.688	50	<0.001	Y	K	H	D	Z	L
dominika (pres.)	0.197	0.663	41.6	<0.001	Z	Y	H	K	L	D
dominika (pref.)	0.2	0.75	42	<0.001	Z	Y	H	K	D	L
landSajfa (pres.)	0.182	0.663	39.6	<0.001	Y	H	K	Z	D	L
landSajfa (pref.)	0.298	0.788	55.2	<0.001	Y	H	K	Z	D	L
oilColors (pres.)	0.285	0.65	53.6	<0.001	Y	Z	K	H	D	L
oilColors (pref.)	0.209	0.75	43.2	<0.001	Y	K	Z	H	D	L
rainyDay (pres.)	0.025	0.588	18.4	>0.2	H	Z	K	D	Y	L
rainyDay (pref.)	0.159	0.688	36.4	<0.002	K	H	Y	D	Z	L
rooster (pres.)	0.090	0.588	27.2	<0.05	H	Z	K	Y	D	L
rooster (pref.)	0.105	0.575	29.2	<0.2	Z	H	Y	D	K	L
butterfly (pres.)	0.28	0.725	52.8	<0.001	K	D	Z	Y	H	L
butterfly (pref.)	0.179	0.613	39.2	<0.001	H	Y	D	K	Z	L
cartoon (pres.)	0.410	0.9125	70.4	<0.001	Y	H	Z	D	L	K
cartoon (pref.)	0.161	0.738	36.8	<0.002	Y	H	K	Z	D	L
clothing (pres.)	0.209	0.788	43.2	<0.001	H	Y	D	Z	K	L
clothing (pref.)	0.144	0.663	34.4	<0.1	Y	H	K	D	Z	L
conver. (pres.)	0.236	0.575	46.8	<0.001	K	Z	H	D	Y	L
conver. (pref.)	0.298	0.75	55.2	<0.001	Z	Y	H	D	K	L
painting (pres.)	0.25	0.863	48.8	<0.001	Y	D	Z	H	K	L
painting (pref.)	0.23	0.788	46.4	<0.001	Z	D	Y	H	L	K
ramp (pres.)	0.197	0.8	41.6	<0.001	H	K	Y	D	L	Z
ramp (pref.)	0.073	0.675	24.8	<0.1	H	K	D	L	Y	Z
spiral (pres.)	0.283	0.65	53.2	<0.001	H	Y	Z	K	D	L
spiral (pref.)	0.105	0.6	29.2	<0.02	H	K	Y	Z	D	L
text (pres.)	0.404	0.875	69.6	<0.001	H	K	Y	Z	D	L
text (pref.)	0.084	0.738	26.4	<0.05	H	K	Y	D	Z	L

Tabuľka 5.1: Výsledky pre každé video. Skratky:pres.=presnosť, pref.=preferencia, priem.=priemer, st.v.=stupne voľnosti, Y=CIE Y, D=Decolorize07, H=Hu14, K=Kim09, L=Lee10, Z=Zhao15. Metódy sú radené zľava od najlepšej po najhoršiu.

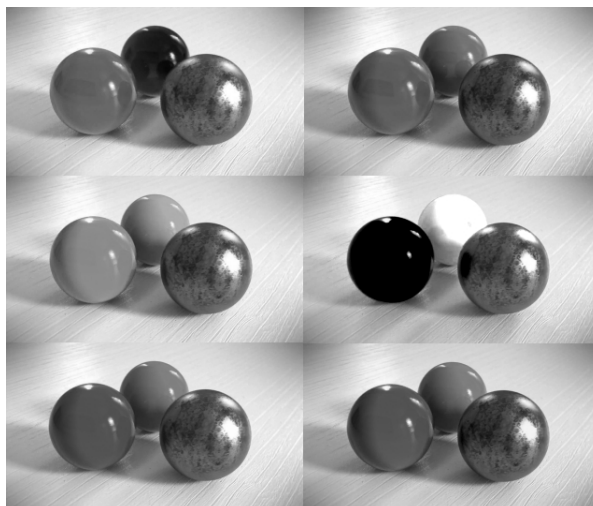
5.3.1 Výber zaujímavých výsledkov

Prvým zaujímavým výsledkom je video *dog* (tab. 5.1). Toto video má veľmi malý koeficient zhody u a pomerne malý koeficient konzistencie ζ . Zároveň sa nepodarilo vyvrátiť nulovú hypotézu H_0 ani v jednom z testov čo značí, že medzi účastníkmi nie je žiadna zhoda. Dôvodom môže byť podobnosť výsledkov obr. 5.2 jednotlivých operátorov. Táto podobnosť sťažuje účastníkom rozhodovanie v obidvoch testoch. Nižší koeficient ζ taktiež môže naznačovať podobnosť výsledkov (2.6.3). Pri teste s referenciou by sme očakávali lepší výsledok ako pri teste bez referencie no v tomto prípade pravdepodobne zohráva svoju úlohu aj referenčné video ktoré obsahuje málo farieb takže výber bol pomerne ťažký aj s týmto videom. Toto zrejme viedlo k tomu, že aj referenčný test mal tiež nižšie hodnoty koeficientov. Pri pohľade na graf *dog* v prílohe A môžeme vidieť, že koeficienty $r = 0.66$ a $R^2 = 0.44$ zobrazujú pomerne slabú koreláciu. Tieto koeficienty boli takisto ovplyvnené tým, že videá si boli podobné a užívatelia sa ťažko rozhodovali.



Obr. 5.2: Zobrazenie 50. snímku odfarbeného 6 metódami ktoré boli použité v teste. Začínajúc vľavo hore po riadkoch sú operátory zoradené nasledovne: Decolorize, Hu14, Kim09, Lee10, CIE Y, Zheng.

Pri videách *balls*, *bunny*, *rooster*, *clothing* a *ramp* sa nepodarilo vyvrátiť nulovú hypotézu v prípade testu zameraného na preferenciu. V teste zameranom na preferenciu majú malý koeficient zhody u no koeficient konzistencie je pomerne vysoký. Toto všetko by nám naznačovalo, že videá boli od seba rozlíšiteľné ale účastníci sa nezhodli len na základe subjektívnosti. Pri pohľade na výsledky týchto videí pri teste s referenciou si všimneme zlepšenie vo všetkých koeficientoch čo len potvrdzuje to, že videá boli od seba aspoň čiastočne rozlíšiteľné. Týmto možno povedať, že dôvodom malého koeficientu zhody pri preferenčnom teste a tým pádom nevyvrátenia nulovej hypotézy je, že videá boli síce rozlíšiteľné ale nie natolko aby sa dokázali účastníci dokázali zhodnúť bez referencie (obr. 5.3). Pri pohľade na grafy v prílohe A si môžeme všimnúť pomerne dobré hodnoty korelačného koeficientu r ktoré sa pohybujú od 0.8 do 0.9 u videí *bunny*, *clothing*, *ramp* a *rooster* ktoré by mohli naznačovať koreláciu medzi testami. Pri videu *balls* je hodnota $r = 0.56$ čo znamená, že nie je žiadna korelácia medzi testami.

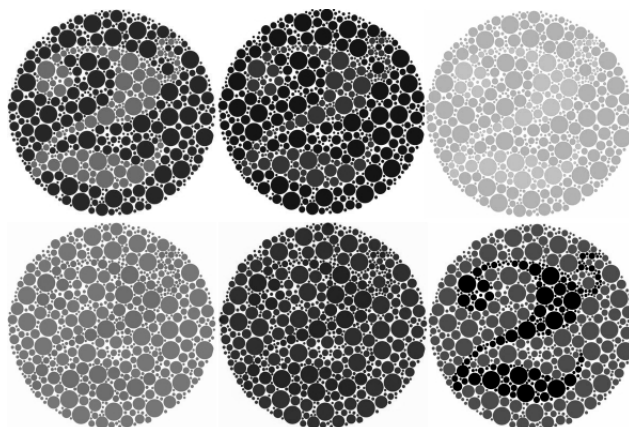


Obr. 5.3: Zobrazenie 71. snímku odfarbeného 6 metódami ktoré boli použité v teste. Začínajúc vľavo hore po riadkoch sú operátory zoradené nasledovne: Decolorize, Hu14, Kim09, Lee10, CIE Y, Zheng.

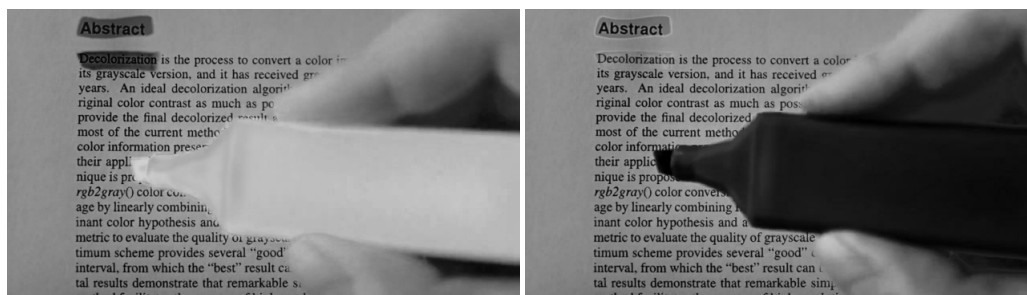
Videám *rainyDay* a *action* sa nepodarilo vyvrátiť nulovú hypotézu pri teste zameranom na presnosť. Je zaujímavé, že účastníci sa nezhodli pri teste s referenciou ale pri preferenčnom teste nastal istý stupeň zhody. Toto by mohlo byť spôsobené pravdepodobne tým, že pri teste s referenciou každý účastník považoval zachovanie inej charakteristiky z referenčného videa za dôležité.

Pri videách *25_vid* a *colorWheel* vidíme jedny z najväčších koeficientov u v rámci datasetu a takisto jedny s najväčších koeficientov ζ v rámci datasetu. Toto poukazuje na to, že videá boli od seba do určitej miery odlišné či už tým, že dokázali video odfarbiť zmysluplne alebo na videu zlyhali vid [5.4](#). Zároveň však treba pouvažovať nad tým či zlyhanie niektorých metód na týchto dvoch videách nemalo za následok to, že sa zväčšila zhoda medzi používateľmi.

Z tabuľky [5.1](#) takisto vyplýva, že metóda Hu14 je najuniverzálnejšou metódou keďže nebola ani raz na poslednom mieste a väčšinou sa držala v najlepšej trojici metód. Metóda Kim09 podávala kolísavé výkony v závislosti od typu videa. Každá jedna metóda sa aspoň raz vyskytla na prvom mieste čo by len potvrdzovalo súvislosť medzi výkonom metódy a typom vstupného videa. U metódy Decolorize sa v niektorých videách prejavila temporálna inkoherencia obr. ([5.5](#)), čo malo dopad na jej hodnotenie no napriek tomu dokázala u určitých typoch videí byť na popredných priečkach. Je zaujímavé, že referenčná metóda CIE Y skončila na prvom mieste hneď niekoľko krát. Metóda Lee10 sa poväčšinou umiestňovala na posledných priečkach avšak vo videu *25_vid* bola zaznamenaná na prvej priečke. Je to pravdepodobne kvôli tomu, že táto metóda ponúkala najkontrastnejší výsledok (obr. [5.4](#)).



Obr. 5.4: Zobrazenie 1. snímku odfarbeného 6 metódami ktoré boli použité v teste. Začínajúc vľavo hore po riadkoch sú operátory zoradené nasledovne: Decolorize, Hu14, Kim09, CIE Y, Zheng, Lee10.



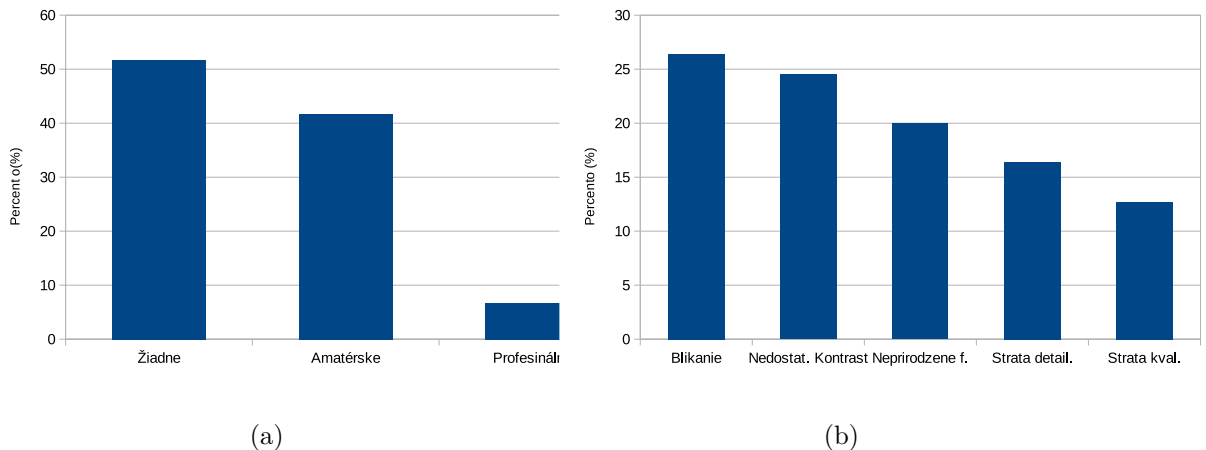
snímok č.163

snímok č.163

Obr. 5.5: Temporálna inkoherencia pri metóde Decolorize

5.4 Výsledky dotazníka

Z dotazníka na konci testu vyplynulo, že sa ho zúčastnilo celkovo 19 žien a 41 mužov a vekové rozpätie sa pohybovalo od 17 do 56 rokov. Na otázku ohľadom farbosleposti boli všetky odpovede negatívne, takže nikoho údaje nemuseli byť z testu vynechané. Ďalej boli zisťované skúsenosti so spracovaním obrazu medzi účastníkmi ktoré dopadli tak, že väčšina účastníkov nemala žiadne alebo len amatérske skúsenosti (obr. 5.6a). Toto nám napovedá, že títo účastníci by nemali mať nejaké konkrétne očakávania pred vstupom do testu a nevedia na čo konkrétne sa majú sústrediť. Druhá otázka zameraná na rušivé faktory pri videu ukazuje, že najviac vyrušuje blikanie videa a nedostatočný kontrast vid 5.6b. Takýto výsledok bol viac menej predpokladaný na základe pilotnej štúdie.



Obr. 5.6: Na grafe (a) sú skúsenosti so spracovaním obrazu, na grafe (b) sú uvedené rušivé faktory pri sledovaní videa.

Testy boli rozoslané známym na sociálnych sieťach s tým, že môžu testy ľubovoľne šíriť. Ďalej boli testy uverejnené v nasledujúcich skupinách na sociálnej sieti: VUT FIT MIT 2016-2018², PPV - Koleje pod Palackého vrchem (VUT Brno)³, VUT FIT BIT 2016-2019⁴, VUT FIT MIT 2015-2017⁵. Testy sú dostupné na linku: <http://www.stud.fit.vutbr.cz/~xvlnkov01/index.php>. Po kliknutí na tento link je účastníkovi testu náhodne priradený jeden zo šiestich možných variatov testov 3.1.

5.5 Zhrnutie

V hodnotení celého datasetu bola zistená korelácia medzi testom bez referencie a testom s referenciou. Pri jednotlivých videách bola táto korelácia potvrdená u väčšiny z nich. Zistilo sa, že pridanie referencie má aspoň čiastočný vplyv na výsledky metód ktoré boli o niečo lepšie ako v preferenčnom teste. Porovnanie časovo závislých metód a časovo nezávislých metód nevyznieva v prospech ani jednej strany. Najlepšie hodnotená metóda Hu14 je síce časovo závislou a jej celkový výsledok by sa dal zaradiť k tým suverénnejším avšak druhá časovo závislá metóda metóda Kim09 už nemala také dobré výsledky. Táto metóda vo veľkej miere závisela od vstupného videa. Z výsledkov nezávislých metód by sa dalo povedať, že pokiaľ časovo nezávislá metóda neprodukuje nejaké artefakty a dodržiava tým pádom temporálnu koherenciu tak môže byť konkurencieschopná v rámci časovo závislých metód. Za ako veľmi prekvapivé možno označiť to, že metóda CIE Y dokázala niekoľko krát predčiť nie len metódu Hu14 ale i všetky ostatné metódy. Z týchto faktov môžeme usúdiť, že výber vstupného videa má aspoň čiastočný vplyv na výsledné poradie metód. Z dotazníka bolo zistené, že účastníkov na videách najviac rušilo blikanie videa pri určitých metódach a chýbajúci kontrast..

²https://www.facebook.com/groups/1064936046887651/?ref=group_browse_new

³https://www.facebook.com/groups/393802530659892/?ref=group_browse_new

⁴https://www.facebook.com/groups/fitbit2016/?ref=group_browse_new

⁵https://www.facebook.com/groups/1650714478496508/?ref=group_browse_new

Kapitola 6

Záver

Cieľom tejto práce bolo navrhnúť metodiku a dataset na otestovanie vlastností časovo závislých metód pre prevod videa do šedotónu a vykonanie používateľského experimentu.

Po naštudovaní problematiky časovo závislých metód pre prevod videa do šedotónu boli naimplementované 3 takéto metódy do systému TMS. V rámci riešenia diplomovej práce bola naimplementovaná knižnica pre načítanie videa do systému TMS. V ďalšom kroku bol vytvorený dataset 37 videí z ktorých sa v teste použilo 24. Dataset bol od začiatku koncipovaný aby bolo umožnené jeho zdieľanie. Na videá v datasete sa nevzťahujú žiadne licencie takže je možné ho voľne šíriť a používať. V nasledujúcom kroku bola navrhnutá metodika testovania videí. Metodika je postavená na párovom testovaní metódou 2AFC. Boli navrhnuté dva testy v rámci metodiky: test bez referenčného videa zameraný na preferenciu a test s referenčným videom ktorý sa sústredil na presnosť prevodu. Na základe datasetu a navrhnutej metodiky bol vykonaný používateľský test ktorého sa zúčastnilo 60 ľudí ktorý vykonali 7200 párových porovnaní videí. Výsledky testu boli analyzované na celom datasete a zároveň pre jednotlivé videá. Pre jednotlivé videá boli vypočítané koeficienty skúmajúce zhodu a konzistenciu v rámci rozhodnutí medzi používateľmi. Bolo analyzované aj poradie jednotlivých metód v každom videu pre každý z dvoch navrhnutých testov. Z testu bolo zistené, že najuniverzálnejšou metódou je časovo závislá metóda Hu14 a že časovo závislá metóda Kim09 nemala takú dobrú výkonnosť ako sa očakávalo. Ukázalo sa však, že časovo závislé metódy nie sú dominantné v porovnaní s nezávislými metódami. Z analýzy poradia metód môžeme usúdiť, že výber vstupného videa má vplyv na výkon metód čím bolo ovplyvnené ich poradie v jednotlivých videách. Takisto bolo zistená korelácia medzi testom bez referencie a testom s referenciou. Ďalším poznatkom je, že test s referenciou má aspoň čiastočný vplyv na výsledky metód ktoré boli o niečo lepšie ako v preferenčnom teste bez referencie. Z dotazníka bolo zistené, že účastníkov testu najviac pri sledovaní videí rušilo ich blikanie a nedostatok kontrastu.

Za pokračovanie práce by som označil implementovanie nových časovo závislých metód. Tieto metódy by mohli byť následne otestované na väčšom počte používateľov ktorý by vedeli poskytnúť reprezentatívnejšie výsledky.

Literatúra

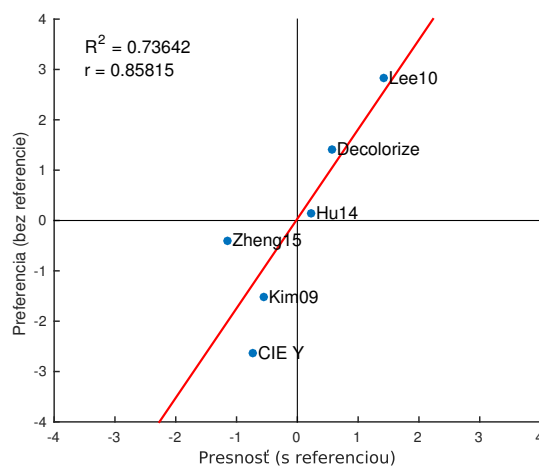
- [1] Denis Pielli, B. F.: Psychophysical methods. In *Handbook of Optics*, editace D. R. W. M. Bass, E. W. Van Stryland, kapitola 29, New York: McGraw-Hill, 1995.
- [2] Dufaux, F.; aj.: *High Dynamic Range Video: From Acquisition, to Display and Applications*. Academic Press, 2016, ISBN 978-0-08-100412-8.
- [3] Gabriel Eilertsen, R. M., Robert Wanat: Evaluation of Tone Mapping Operators for HDR-Video. *Computer Graphics Forum*, ročník 32, č. 7, 2013: s. 275–284.
- [4] Hanli Zhao, X. J., Haining Zhang: Efficient image decolorization with a multimodal contrast-preserving measure. *Computers & Graphics*, ročník 70, 2018: s. 251 – 260, cAD/Graphics 2017.
- [5] Hu, W.; aj.: Real-time Decolorization using Dominant Colors. [Online; navštíveno 10.10.2017].
URL <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.727.6443&rep=rep1&type=pdf>
- [6] Intel: Color Models. [Online; navštíveno 29.4.2018].
URL <https://software.intel.com/en-us/ipp-dev-reference-color-models>
- [7] Jiří Žára, J. S. a. j., Bedřich Beneš: *Moderní počítačová grafika*. Computer Press, 2004, ISBN ISBN 80-251-0454-0.
- [8] Joseph Lee Rodgers, W. A. N.: Thirteen Ways to Look at the Correlation Coefficient. *The American Statistician*, ročník 41, č. 1, 1988: s. 59–66.
- [9] Kim, Y.; aj.: *Robust Color-to-gray via Nonlinear Global Mapping*. *ACM Transactions on Graphics (SIGGRAPH ASIA 2009)*, ročník 28, č. 5, 2009.
- [10] Mark Grundland, N. A. D.: Decolorize: fast, contrast enhancing, color to grayscale conversion. *Pattern Recognition*, ročník 40, č. 11, 2007: s. 2891–2896.
- [11] Minolta, K.: Understanding the CIE L*C*h Color Space. [Online; navštíveno 29.4.2018].
URL <https://sensing.konicaminolta.us/blog/understanding-the-cie-lch-color-space/>
- [12] Montgomery, D. C.: *Applied statistics and probability for engineers*. John Wiley & Sons, Inc., 2003, ISBN 0-471-20454-4.

- [13] Patric Ledda, A. C.: Evaluation of Tone Mapping Operators using a High Dynamic Range Display. *ACM Transactions on Graphics (TOG)*, ročník 24, č. 3, 2005: s. 640–648.
- [14] Pospíšil, P.: *Převod barevných obrázků na černobílé*. Diplomová práce, Vysoké učení technické v Brně, Fakulta informačních technologií, Brno, 2016.
- [15] Rafal Mantiuk, J. P.: Assessment of video tone-mapping: Are cameras' S-shaped tone-curves good enough? *Journal of Visual Communication and Image Representation*, ročník 24, č. 7, 2013: s. 1020–1030.
- [16] Tae-Hee Lee, W.-J. S., Byoung-Kwang Kim: Converting color images to grayscale images by reducing dimensions. *Optical Engineering*, ročník 49, č. 5, 2010.
- [17] Tao, Y.; Shen, Y.; Sheng, B.; aj.: Video Decolorization Using Visual Proximity Coherence Optimization. *IEEE Transactions on Cybernetics*, ročník 48, č. 5, 2018: s. 1406–1419, publikované v 2017.
- [18] Walter Ehrenstein, A. E.: Psychophysical methods. In *Modern Techniques in Neuroscience Research*, editace J. H. Windhorst U., kapitola 43, Berlin: Springer, 1999, s. 1211–1241.
- [19] Zheng, X.; Feng, J.; Zhou, B.: Efficient Color-to-gray Conversion for Digital Images in Gradient Domain. *Proceedings of the 14th ACM SIGGRAPH International Conference on Virtual Reality Continuum and Its Applications in Industry*, 2015: s. 85–88.
- [20] Čadík, M.: Perceptual Evaluation of Color-to-Grayscale Image Conversions. *Computer Graphics Forum*, ročník 27, č. 7, 2008: s. 1745–1754.
- [21] Ševčík, V.: *Webový nástroj pro párový test obrázků*. Bakalářská práce, FIT VUT v Brně, Brno, 2015.

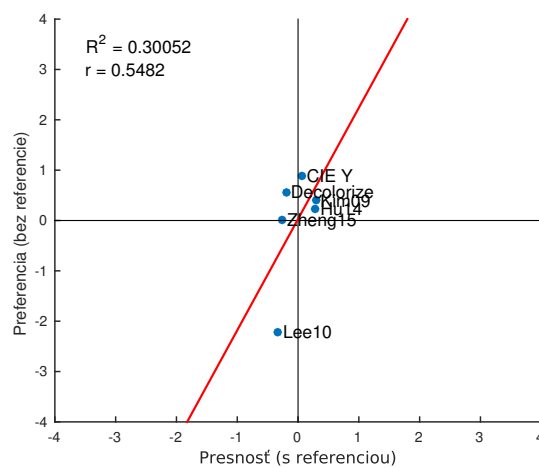
Príloha A

Grafy

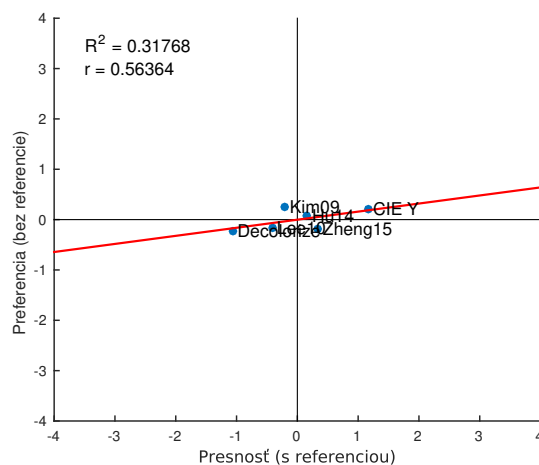
Grafy porovnávajúce presnosť a preferenciu pre každé video z datasetu.



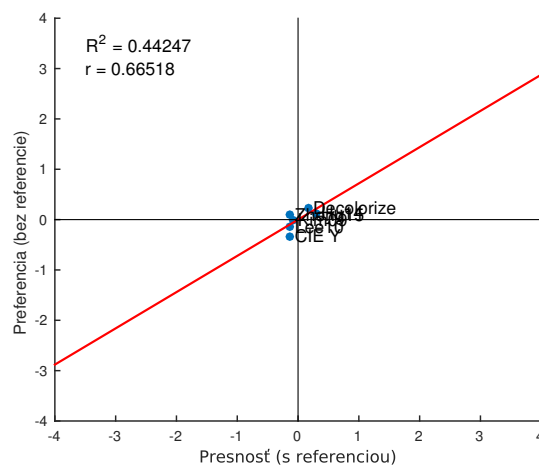
25_vid



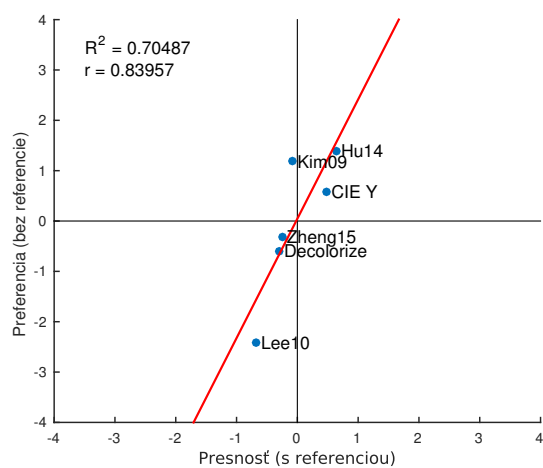
action



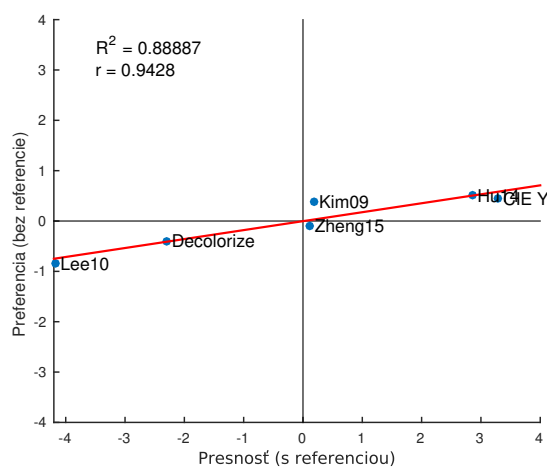
balls



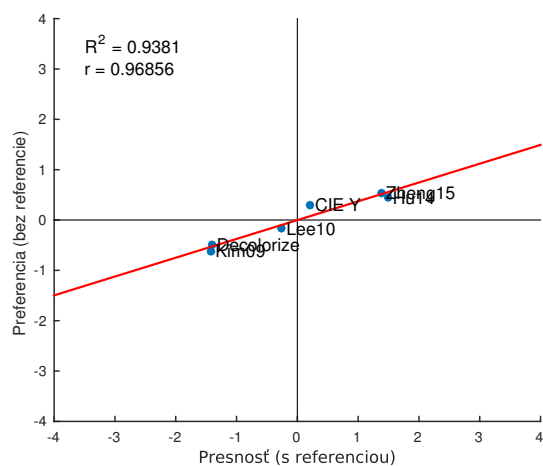
dog



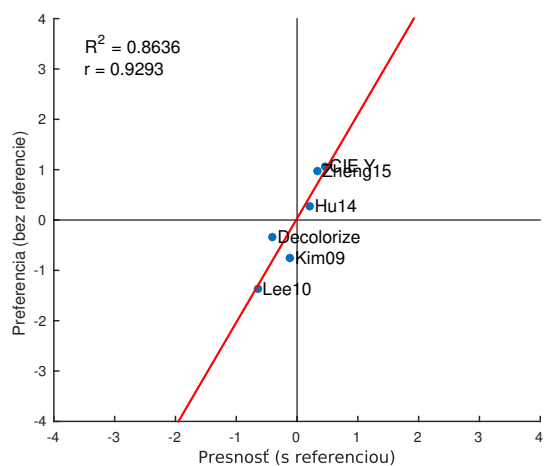
portrait



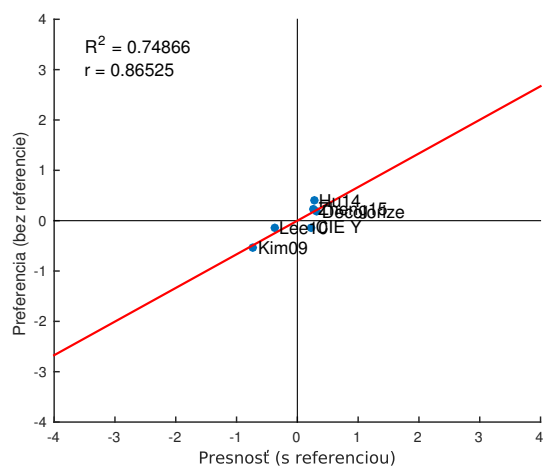
red



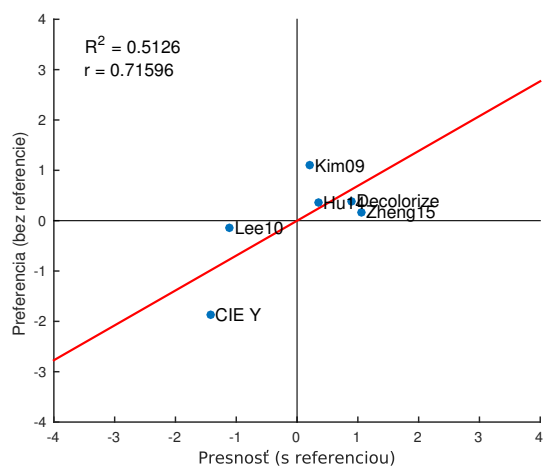
scandinavia



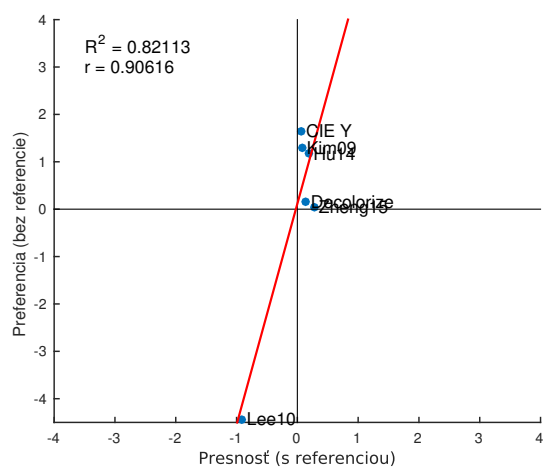
tetris



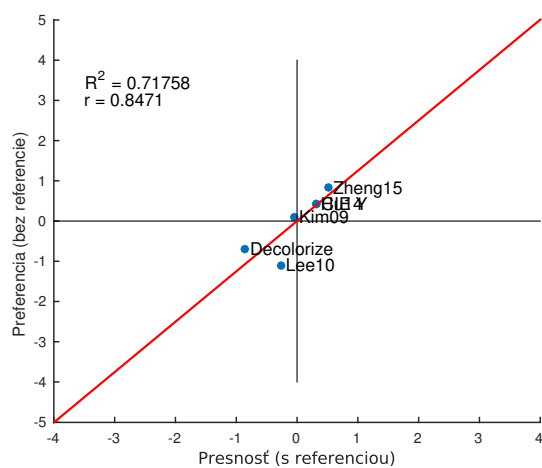
bunny



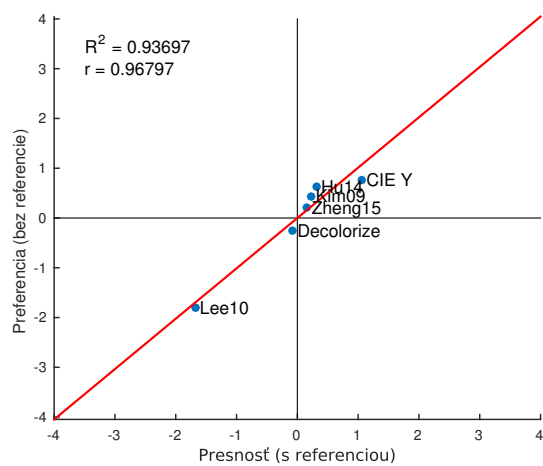
colorWheel



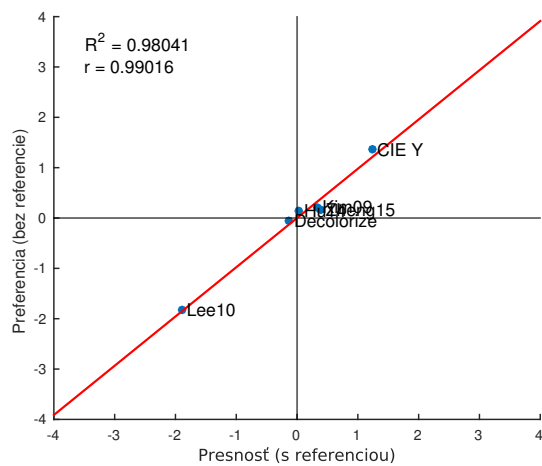
cooking



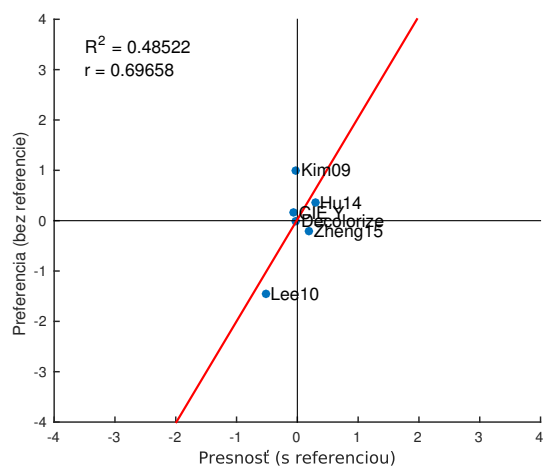
dominika



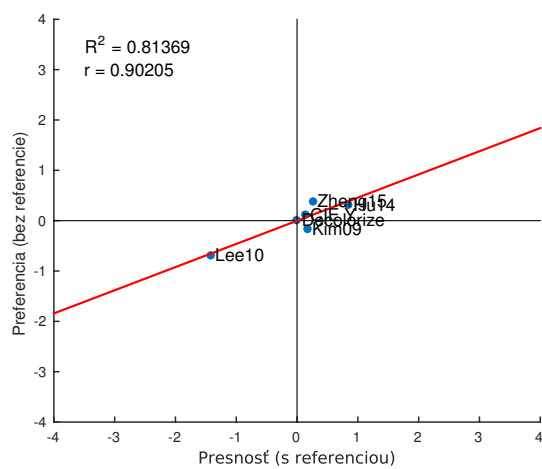
landscapeSajfa



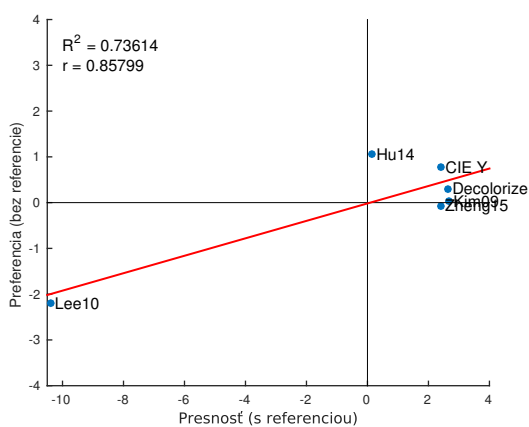
oilColors



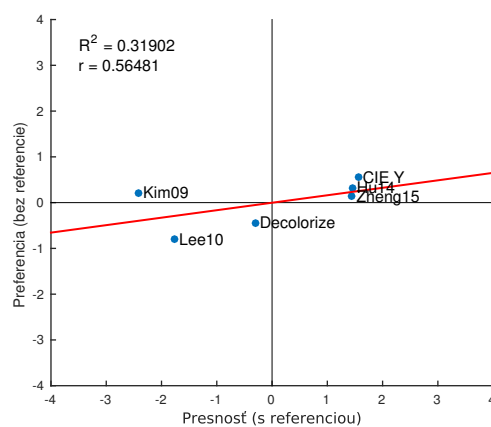
rainyDay



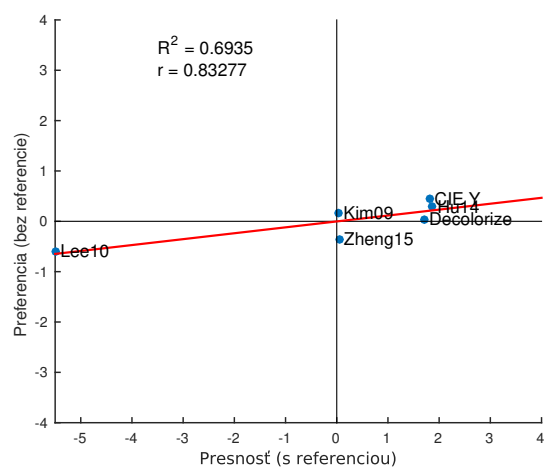
rooster



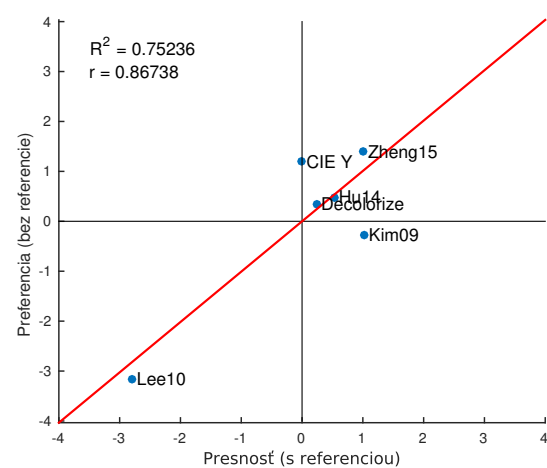
butterfly



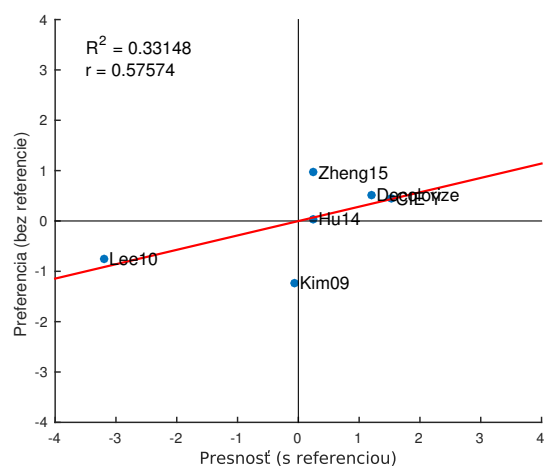
cartoon



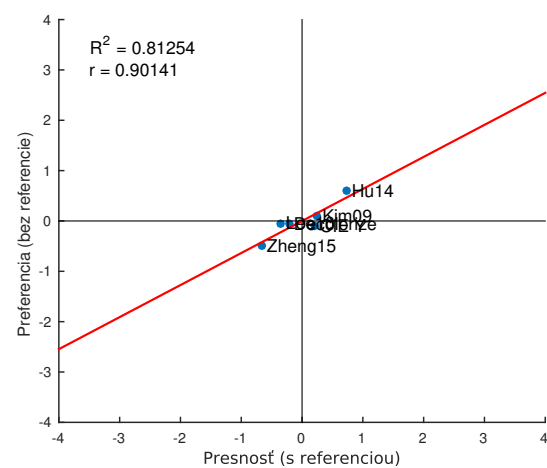
clothing



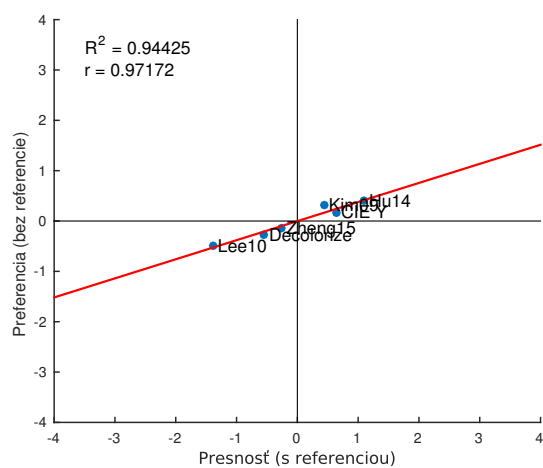
conversation



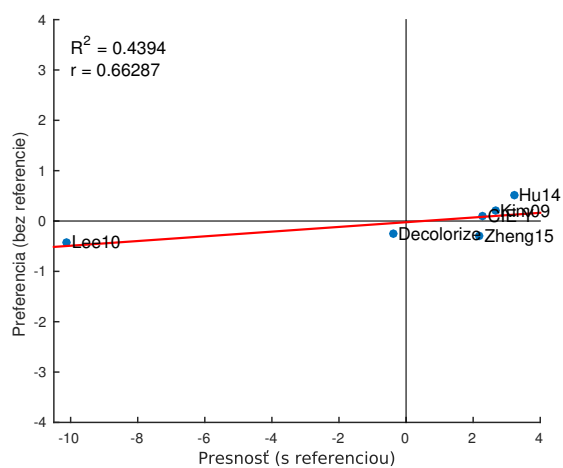
painting



ramp



spiral



text

Príloha B

Odkazy na videá

Odkazy na videá z ktorých bol vytvorený dataset. sajfa <https://www.youtube.com/watch?v=HMAPJDsUj8E>
scandinavia <https://pixabay.com/en/videos/scandinavia-arctic-landscape-coast-1410/>
fruit salad <https://pixabay.com/en/videos/fruit-salad-bowl-fruits-fruit-1019/>
red <https://pixabay.com/en/videos/red-flower-summer-color-colorful-14840/>
rooster <https://pixabay.com/en/videos/cock-chicken-village-farm-polygamy-10685/>
ski <https://pixabay.com/en/videos/ski-winter-snow-mountains-8558/>
dog <https://pixabay.com/en/videos/dog-drinking-pet-food-thirsty-5631/>
book <https://pixabay.com/en/videos/book-write-pen-diary-letter-hands-4952/>
light effect <https://pixabay.com/en/videos/light-effect-disco-ball-party-club-3129/>
clothing <https://pixabay.com/en/videos/clothing-clothes-fashion-shop-1006/>
portrait <https://pixabay.com/en/videos/portrait-kids-wreath-flowers-smile-11244/>
rainy day <https://pixabay.com/en/videos/rainy-day-rain-night-summer-5275/>
oilColours <https://pixabay.com/en/videos/colours-macro-green-universe-oil-14506/>
pinwheel <https://pixabay.com/en/videos/wind-power-wind-energy-11051/>
swan <https://pixabay.com/en/videos/swan-bird-evening-twilight-1646/>
trees <https://pixabay.com/en/videos/trees-forest-cemetery-cologne-12768/>
red blur <https://pixabay.com/en/videos/red-light-background-light-leak-4865/>
spiral <https://pixabay.com/en/videos/spiral-hypnosis-rotation-color-4087/>
balls <https://pixabay.com/en/videos/balls-about-blender-3d-red-blue-5875/>
not ludo <https://pixabay.com/en/videos/not-ludo-board-game-7118/>
cooking <https://pixabay.com/en/videos/cooking-mackerel-fish-food-healthy-11635/>
butterfly <https://pixabay.com/en/videos/peacock-butterfly-butterfly-nature-4049/>
girls <https://pixabay.com/en/videos/girls-pedestrian-crossing-city-6575/>
rollercoaster <https://videos.pexels.com/videos/roller-coaster-852415>
painting <https://videos.pexels.com/videos/painting-and-drawing-857115>
conversation <https://mango.blender.org/download/>
action <https://mango.blender.org/download/>
bunny <https://peach.blender.org/download/>
cartoon <https://www.youtube.com/watch?v=lqiN98z6Dak&index=3&list=PL6B3937A5D230E335>